Counselling Portal
(AskAkak.com)

By

Latifah Bte Abd Latif
Wek 990268

Under the supervision of

Dr. P. Sellappan

Moderated by

Prof Madya Puan Raja Ainon Zabariah

Raja Zainal Abidin

A Project Submitted to the
Faculty of Computer Science and Information Technology
University of Malaya
Session 2002/2003

Acknowledgement

First and foremost, I would like to express my deepest gratitude and thanks to my project supervisor, Dr. P. Sellappan, for his invaluable assistance, ideas, advice and utmost guidance throughout this project development.

I would also like to express my gratitude to my moderator, Assoc. Prof. Pn. Raja Ainon Zabariah Bte Raja Zainal Abidin for her comments and suggestions on the system.

Besides I would like to present my utmost gratitude to my parent, Abd Latif Bin Abu Bakar and Induk Ungah Bte Hj Daing Melebik, my brothers, Zainal Abidin Bin Abd Latif, Mohd Jamal Bin Abd Latif and Mohd Faizal Bin Abd latif and lastly my sister, Norshahidah Bte Abd Latif, who always supported me in whatever I'm undertaking.

My heartfelt thanks also go to my friends Nor Azlila and Khairiah. Our coordination was a first success in order to complete this thesis successfully.

Last but not least, much appreciation is attributed to all my fellow course mates, for their willingness to share a lot of opinions and experiences throughout the project development.

Abstract

Throughout our lives, we encounter many challenges like conflicts in relationship, life transaction, emotional crisis, educational and career decisions. Sometimes the challenging are interested and sometimes there are frightening. And sometime we can be benefit from help and support as we strive to master them.

Who is qualified to provide counseling portal? There is confusing issue for those who are seeking services. Today, there are a lot of portal offering services particularly in psychology from online counseling, e – counselor, motivation to behavioral analysis. The responses from user are overwhelming for those kind of online services. However, the current portal counseling systems are mostly advertisement oriented. This is because, most of the website is playing role as a medium to advertise a particular physician business.

Therefore, we develop this portal to help especially for the university students to discuss and solve their problems such as academic, career and relationship problems. Besides it is free, the students can share their problems among themselves in forum and chat room, which are provided in our portal. This portal has two sections, which are user and administrator section. The user section allows users browse through all the services that the system provides like chat room, forum and etc. The administrator section provides an application for the administrator to update the counseling info in the database.

Table of Contents

List of Figures

CHAPTER 1
Introduction

Project overview

From childhood through late adulthood, there is certain timer when we may need helps addressing problems and issues that cause us emotional distress and make us feel overwhelmed. Globally, counselling - an interpersonal helping relationship which begin with the clients exploring the way they think, how they feel for the purpose of enhancing their life. The clients determines and declares to the counsellor what the counter productive behaviors are and then make decision about which one will be work done. The counsellor helps the clients to set the goals.

Counseling has many different scopes, which is divided into 6 main approaches:-

1. **Person – centered**

   A way of working in counseling that puts the client at the center of the activity, and has the belief that the client knows best how to solve their problems.

2. **Eclectic**

   Any counseling theory or practice that uses and combines beliefs, findings and techniques selected from a wide range of theoretical systems.

3. **Psychodynamic**

   A general term for approaches to counsel heavily influenced by the work of Sigmund Freud.

4. **Transactional Analysis**

   Therapeutic work with individuals and their relationship originating in the work of Eric Berne.

5. **Gestalt**

   A humanistic psychotherapy approach based on the work of Fritz Perls.

6. **Psychotherapy**

   Psychotherapy means working on problems using psychological methods. Depending on the question, the approach used may be through counseling, listening, relaxation, dream – work, fantasy work, story telling and many other techniques, all with aim of assisting your deeper self to help you to achieve beneficial change.

Thus counseling thru out the scope has many different criteria, services, dimension, theories and guidelines.

As we know there are many organizations, schools, educational enters and higher learning institutes offers various counseling help. In accordance to the objectives these organizations are also involve in many programs to develop a very good relationship to solve and help those who needs help.

Definition

A counseling portal is web – based informative application. It provides very useful information on counselling aspects for the users especially youths in academic, career and relationship issues. This will be guidance to the youths to know what are the proper guidelines and supportive assistance they could find in solving their problems.

The user can retrieve the information on their issues, provided assistance, answers, required information and common questions related accordingly to the issues, guidance of the professional counsellors and a lot more. User can get many information directly by clicking the appropriate icons. Forum section will be provided where users can interact with the counselors to discuss further about their problems.

This will be more useful and effective to the users in order to minimized their time, cost and others, which could be easily, handle without much procedures and work limit.

Project motivation

Undoubtedly Internet has emerged as an effective communication channel in the late 20 century. With the introduction of browser such as Mosaic and Netscape, which features graphical user interfaces, the WWW has become readily accessible.

Organization, universities and schools rushed to become wired in order to provide information to of Internet usage among users.

This project has been suggested to study and develop a COUNSELLING PORTAL. In order to assists the target users especially the university students to tackle and overcome their problems, based on the different criteria of the issues. These criteria, for example:

- Students problems like love, abuse, drug
- Academic problems like tackling exam, being interviewed
- Career problems like career depression, competing with others for higher positions .

Based on the preliminary research, current counseling portal in global does not have suitable solutions to help or assist students to reconstruct and develop till the maximum of their capital.

This project is definitely feasible for implementation with the following reasons:

1) Internet has become a cheap mean of publicity and disseminating information. Nowadays, we can get access as low hardware cost. Therefore, it is wise to develop a portal for counseling information system for students.

2) Information can be acquired through and solutions in different prospect could be solved immediately.

Objective

- Develop & provide search by displaying solution to problems. Usage of search algorithm towards the improvement of an accurate search.
- Helps in overcoming the shortage of professional counselor in our country.
- Giving professional help through the web which free and easier access for everyone.
- Create a counseling environment to attract counselees to the site.
- Create a counselor's circle that can interact with the counselees.
- Secured way of communications.

Project goal

As an alternative to the traditional face – to – face counseling

- **Waste of time**
  - process making appointments and finally meeting, the counselor can be long and unbearable.
  - Sometimes counselor see limited and notable to attend to the need of students.

- **Difficulties in receiving immediate advice**
  - when counselees are facing problem they can't receive counseling immediately
  - counselor not available
  - can 'battle' the students

- **location**
  - some people might find in difficult for students who are staying far for counseling center to receive immediate help.

- **Inflexible time**
  - operating hour of counseling units are always during office hours and overlapping with students timetable.

To create a better counseling portal than the existing one in term of

- **user friendliness**
  - we do not use complex web design consistent
- **the portal itself**
  - it is a complete, through and comprehensive set of in information
- **easy access to information**
  - user does not have to dig through because information can be easily captured.
- **It deals with 3 types of counseling at once**
  - academic counseling
  - career counseling
  - relationship counseling

Project Expected Strengths

The current Portal counseling give rises to certain problems. This portal is expected to overcome some of these problems through some of its strengths as follows:

1) Clients or users can get free counseling from professional counselors.

2) Clients or users can get the counseling services or information from anywhere, anytime as long as they have the Internet access to our portal.

3) Clients can discuss their problems among themselves or with counselors.

4) User – friendly Interface

   The system will have a user – friendly interface that will make it easy to use and understand. GUI components such as command buttons and menus will be used to help users navigate through the portal.

5) Relatively Fast Response Time

   The system will be designed in such manner that its' portal are loaded in a reasonable amount of time.

Expected outcome

This project should develop an interactive portal counseling system that provides useful and effective counseling information. The system can receive comments and suggestions from the user to enhance the system. The system also provides real time info exchange function between users with users or users with counselors. The system also provides a backend application for the administrator and counselors to maintain and control the system.

Project Schedule

Activities	Mar 2002	Apr 2002	May 2002	June 2002	July 2002	Aug 2002	Sep 2002
Preliminary Investigation Phase ▪ Design objectives							
Literature Review							
System Analysis Phase ▪ Study current software ▪ Improvements ▪ Prepare the new model							
System Design Phase ▪ Study available guidelines ▪ User defined interface							
System Development Phase ▪ Coding							
Testing Phase ▪ Operational software interface ▪ Benchmarking							

Figure 1-1AskAkak.com Project Schedule

CHAPTER 2
Literature Review

Literature Review

According to Cooper (1988) '...a literature review uses as its database reports of primary or original scholarship, and does not report new primary scholarship itself. The primary reports used in the literature may be verbal, but in the vast majority of cases reports are written documents. The types of scholarship may be empirical, theoretical, critical/analytic, or methodological in nature. Second a literature review seeks to describe, summarize, evaluate, clarify and/or integrate the content of primary reports.'

The review of relevant literature is nearly always a standard chapter of a thesis or dissertation. The review forms an important chapter in a thesis where its purpose is to provide the background to and justification for the research undertaken. Bruce, who has published widely on the topic of the literature review, has identified six elements of a literature review. These elements comprise a list; a search; a survey; a vehicle for learning; a research facilitator; and a report.

A crucial element of all research degrees is the review of relevant literature. So important is this chapter that its omission represents a void or absence of a major element in research. According to Bourner (1996) there are good reasons for spending time and effort on a review of the literature before embarking on a research project.

These reasons include:

- To identify gaps in the literature
- To avoid reinventing the wheel that can prevent from making the same mistakes as others
- To identify other people working in the same fields
- To increase breadth of knowledge of the project subject area
- To identify opposing views
- To identify information and ideas that may be relevant to the project
- To identify methods that could be relevant to the project

As far as the literature review process goes, ultimately the goal for students is to complete their review in the allocated time and to ensure they can maintain currency in their field of study for the duration of their research (Ref. 6).

To achieve the above objective, several steps are adapted (Ref.7). The steps are as follows:

1. Be organize around and related directly to the thesis that is going to be developed
2. Synthesize results into summary of what is and is not know
3. Identify areas of controversy in the literature
4. Formulate questions areas that need further research
5. Produce a report base on all the findings

In general, this project literature review will be focused on topics that are relevant to the project objectives. The review will cover topics below:

1. Software development model for multimedia software
2. Analysis of existing relevant software
3. Interactive multimedia software as a medium for education
4. Development tools for multimedia software

Techniques on Conducting Literature Review

Information is essential to do a good research or analysis. For this project, several techniques have been taken to seek information. These techniques are as follows:

- **Refer to reference books from the library**

A lot of references on conference, workshop, journal and symposium can be found from the library. References book on methodology and system design also can be found from the library.

- **Search information from the Internet**

Internet is the main source of information. Relevant information on portal application, client – server and programming tools are analyzed.

- **Do analysis on the past year thesis**

Several past years thesis documentation have been studied in order to identify any potential mistakes and gain some skills on software development.

- **Refer to newspapers and magazines**

Reference on the latest newspaper and magazines such as In Tech (The Star), Computimes (New Straits Times), and PC Magazines is one of the techniques to gain the latest technology for this project.

- **Have discussion with friends and lecturers**

Useful advices have been given for each section meeting conduct with my supervisors and other lecturers. It is useful for errors correctness and act as reminder when carried out the system development process.

- **Conducting interview and survey**

Interview and survey have been conduct with administrator and students to find out the lacks of the current system and potential improvement on it.

Programming Tools

There are so many development tools that can be used to develop good multimedia software. This chapter will cover on what and why are the tools are chosen for the development of this project. The tools that are going to be used are as follows :

1) PHP

PHP is a widely-used general-purpose scripting language that is especially suited for Web development and Before installing first, you need to know what do you want to use PHP for. There are three main fields you can use PHP, as described in the What can PHP do? section:

- Server-side scripting
- Command line scripting
- Client-side GUI applications

For the first and most common form, you need three things: PHP itself, a web server and a web browser. You probably already have a web browser, and depending on your operating system setup, you may also have a web server (eg. Apache on Linux or IIS on Windows). You may also rent webspace at a company. This way, you don't need to set up anything on your own, only write your PHP scripts, upload it to the server you rent, and see the results in your browser. You can find a list of hosting companies at http://hosts.php.net/.

While setting up the server and PHP on your own, you have two choices for the method of connecting PHP to the server. For many servers PHP has a direct module interface (also called SAPI). These servers include Apache, Microsoft Internet Information Server, Netscape and iPlanet servers. Many other servers have support for ISAPI, the Microsoft module interface (OmniHTTPd for example). If PHP has no module support for your web server, you can always use it as a CGI processor. This means you set up your server to use the command line executable of PHP (php.exe on Windows) to process all PHP file requests on the server.

If you are also interested to use PHP for command line scripting (eg. write scripts autogenerating some images for you offline, or processing text files depending on some arguments you pass to them), you always need the command line executable

With PHP you can also write client side GUI applications using the PHP-GTK extension. This is a completely different approach than writing web pages, as you do not output any HTML, but manage windows and objects within them.

What distinguishes PHP from something like client-side JavaScript is that the code is executed on the server. If you were to have a script similar to the above on your server, the client would receive the results of running that script, with no way of determining what the underlying code may be. You can even configure your web server to process all your HTML files with PHP, and then there's really no way that users can tell what you have up your sleeve.

Advantages of PHP

PHP is clearly on the one true language, but some people needs facts; following are facts the clearly show php is scripting language for today websites. Based on experience, PHP is easier to teach than other mixtures, like visual basic and IIS

The best things in using PHP are that it is extremely simple for a newcomer, but offers many advanced features for a professional programmer. Don't be afraid reading the long list of PHP's features. You can jump in, in a short time, and start writing simple scripts in a few hours.

Interpret versus Compile

PHP has a near – perfect blend of compilation and interpretation. It checks about as much as a good compiler will check abd still gives you all the advantages of interpretation.

Parts versus Programming

You can assemble applications out of included files, functuions, objects, and all sorts of code scraps, but none of them replace writing your own code. Lots of free PHP objects and applications are available for download, and many contain deadly coding traps that make them difficult to maintain and unsuitable for a business site.

Regular Expression Functions

PHP supports two types of regular expressions, POSIX and Perl compatible. Perl programmers will be more comfortable with the Perl – compatible regular expression functions include **preg_match(), preg_match_all()** and others.

PHP and Apache

Which web server should you use? Apache or Apache? I have to admit to a bias toward Apache. It is so easy to install, so easy to use, and so reliable, that I recommend Apache first and then discuss the operating system to support it. In the likely event you are stuck with Netscape's Web Server or Microsoft IIS, or any other Web server that is not Apache, you are probably stuck with a whole Web – sit development environment that does not include PHP. Apache is perfect for PHP and perfect for your website.

2) Apache

Apache runs on all the servers. If u do not know which operating system to use, choose Apache so that you are free to change as you wish. Installing Apache on Windows is reliable and easy. PHP installs everywhere Apache installs. The PHP installation details very across platforms, but the parts in Apache relating to PHP are the same.

1. The Apache httpd server
    - is a powerful, flexible, HTTP/1.1 compliant web server
    - implements the latest protocols, including HTTP/1.1 (RFC2616)
    - is highly configurable and extensible with third-party modules
    - can be customised by writing 'modules' using the Apache module API
    - provides full source code and comes with an unrestrictive license
    - runs on Windows NT/9x, Netware 5.x and above, OS/2, and most versions of Unix, as well as several other operating systems
    - is actively being developed
    - encourages user feedback through new ideas, bug reports and patches
    - implements many frequently requested features, including:

DBM databases for authentication

Allows you to easily set up password-protected pages with enormous numbers of authorized users, without bogging down the server.

Customized responses to errors and problems

Allows you to set up files, or even CGI scripts, which are returned by the server in response to errors and problems, e.g. setup a script to intercept **500 Server Error**s and perform on-the-fly diagnostics for both users and yourself.

Multiple DirectoryIndex directives

Allows you to say DirectoryIndex index.html index.cgi, which instructs the server to either send back index.html or run index.cgi when a directory URL is requested, whichever it finds in the directory.

Unlimited flexible URL rewriting and aliasing

Apache has no fixed limit on the numbers of Aliases and Redirects which may be declared in the config files. In addition, a powerful rewriting engine can be used to solve most URL manipulation problems.

Content negotiation

i.e. the ability to automatically serve clients of varying sophistication and HTML level compliance, with documents which offer the best representation of information that the client is capable of accepting.

Virtual Hosts

A much requested feature, sometimes known as multi-homed servers. This allows the server to distinguish between requests made to different IP addresses or names (mapped to the same machine). Apache also offers dynamically configurable mass-virtual hosting.

Configurable Reliable Piped Logs

You can configure Apache to generate logs in the format that you want. In addition, on most Unix architectures, Apache can send log files to a pipe, allowing for log rotation,

hit filtering, real-time splitting of multiple vhosts into separate logs, and asynchronous DNS resolving on the fly.

3) MySQL

Just what is *SQL and MySQL?*

SQL (some people call it "ess Que ell", others insist that it should be pronounce as "sequel") stands for Structured Query Language. SQL is an international standard language for querying databases.

MySQL is, at its root, an *SQL server*, i.e. a program which accepts requests written in SQL and delivers back some type of answer. The answer may be data, the number of records affected by the query, or it may simply be an "ok, did that".

Of course, you must have some way of communicating with MySQL. This may be done either through a client or through your favorite programming language.

What's so great about MySQL?

MySQL is a great database system for **handling very large data sets**. Many users report having tables that contain several hundred thousand, or million, records. Hearing of databases of several Gigabytes are not uncommon on the mailing list.

MySQL **outperforms** many other available systems in querying on large tables. MySQL is also **very stable** and performs well, even when several hundred people are accessing the same data concurrently. This is one of the reasons MySQL is very popular for **web applications**. MySQL also **scales very well**, and runs on anything from small pc's to huge, multi-processor systems.

All that aside, this author thinks that the greatest feature of MySQL is the ability to connect to the same MySQL server, **no matter what operating system** and what programming language or client program you use. Besides a whole bunch of UNIX-type systems, you can run the MySQL server on OS/2 and Windows (see the supported OS types section of the manual).

Among the many programming languages you can use for interfacing to the server are C, C++, Java, PHP, Perl, TCL and Python (check out the feature section of the manual for more information).

Advantages of MySQL

MySQL doesn't support **transactions**. If you don't know what transactions are, chances are you'll never need them. Transactions are most widely used in systems where complex updates of many tables concurrently is important and you must be *absolutely* sure that all things either go right or don't happen at all. That's why banking systems are heavily dependent on transactions. The rest of us, however, can usually live without 'em.

MySQL doesn't support **relational integrity constraints**. To most of us, relational integrity is mostly a convenience that prevent programmers and users from introducing inconsistencies, such as creating orphan records. Usage of relational integrity constraints is often a hazard (personally, I've seen too many things go wrong with a DELETE CASCADE), so the lack of this feature shouldn't upset you terribly, either. For a good explanation of relational integrity (and how to live without it), have a look at the manual entry on foreign keys.

MySQL does not support **sub-selects**, which you use for complex queries. This is being worked on for one of the upcomming releases of MySQL

The are other things missing from MySQL (most notably: triggers, views, stored procedures), but most are up-and-coming features. Have a look at the Functionality missing from MySQL entry in the manual, for further details.

Among the many advantages of this approach is, that the client and the server may be developed independently of each other and that the server is already configured to handle concurrent access from many different users at once. Something which cannot be said of many office applications today, which are mostly oriented toward the single user.

4) Adobe Photoshop 6.0

Adobe Photoshop is one of the industry standards for desktop image editing and graphic manipulation. It provides functions such scanning images, make original art, or composite images as well as performing color correction, retouching and other image manipulations (Ref. 14). Adobe Photoshop 6.0 software introduces the next generation of image editing with powerful new features that offer something for every user. Delivering the broadest and most productive toolset available, Photoshop helps you explore your creativity, work at peak efficiency, and achieve the highest quality results across all media It uses an output sharp type edges with user image to produce high-quality results. It is faster with user-interface enhancements that help you to get to work quickly and make it easy to take full advantage of the comprehensive tool set.

CHAPTER 3
System Analysis

System Analysis

There is need to do system analysis before designing the functional system. It is an important phase in the Software Development Life Cycle (SDLC). System analysis is an attempt to understand the problems and limitations of the existing system. The overall emphasis of the analysis is to gather information of the current system, requirements problems and solutions for the system being developed. The main purposes of the analysis phase are:

- To survey available system to gain some extra understandings for the system being developed.
- To interview with counselors regarding the system requirements to meet the goal of counseling.
- To acquire knowledge on how does counseling works.
- To gain an overall understanding of the system flow and system process.
- To gain appropriate data and requirements.
- To find out the strength and weakness of the system developed.
- To identify the software and hardware to develop the system.

Strength and weakness on existing system

Strength	Weakness
▪ Good control and maintenance for information – the administrator can view, update and delete the information in the system's database. So the data under controlled and ease for maintenance. .	▪ No real time info exchange functional – sometimes, user needs extra information and the information that the system provides is not enough. So a real time function like chat room and forum is needed so that the users can discuss and exchange ideas with each and others.
▪ Easy to search for solutions and principles – the system provide a search engine to seek relevant information instantly.	▪ No direct interaction with counselors – the existing system does not provide any methods for user to interact with the counselors directly.
▪ E – mail notifications – provides e – mail notification functions.	▪ Concentrate only in one topic counseling like career counseling.
▪ Provides report – provides simple and customs made forms to print reports from the portal.	▪ It not deals with 3 types of counseling at once.
▪ Security – provides Login	

module to prevent either unauthorized users to access the system.	

Advantages and Disadvantages of computer counseling to us by John Suler

Advantages	Disadvantages
▪ Task Performance - Computers perform well when given task compared to humans. They do it accurately without missing anything.	▪ Feeling computers do not have feelings like human therefore they can't conduct all therapists.
▪ Personality – it s personality could design to sign the mode of therapy.	▪ Thinking and learning – Limited to change. They do not learn as well as human.
▪ Cost cheaper – because we do not need a professional counselor. It is just one development and installment in the server to host. Maintenance is also cheaper.	▪ Empathy – Computers can have empathy but as well as human.
▪ Accessibility – If it is hosted in	▪ Rapport – Some people feel

the Internet, then it can be accessible at anywhere and anytime.	uncomfortable talking to a computerized therapist.

This chapter will be divided into three major topics, which include:

1) System methodology
2) Requirement analysis
    2.1) Functional requirements
    2.2) Non Functional requirements
3) System requirements
    3.1) Software requirements
    3.2) Hardware requirements

System methodology

Many large organizations have 2 to 5 years application backlog, which is, applications needed to be developed but have yet reached the end – user [Simon Holloway, 1989]. Baldock [R. Baldock, 1984] summarized that the rate at which software can be produced has not kept pace with market needs, hence resulting in an increase in the application backlog.

As such, an effective framework of carrying out development task needs to be in place. Methodology is the practice of methods and procedures in developing a system [Object Agency, Inc., 1992 – 1993]. In this chapter, I will discuss the benefits of having a good methodology and propose a methodology for this project.

Why methodology?

Over the years, many system development methodologies have evolved. A system development methodology does not just provide a set of modeling techniques, it also defines the stages of a system development project, specifies the task to be carried in and out and the input expected from each stage, provides guidelines for project management and control, and is backed by a philosophy on its approach towards system development [Barbara Robinson and Mary Prior, 1995].

A good methodology for the effective way of doing things is best defined before the project starts and becomes he framework to development staff. Below describes a numerous of benefits offered by a good methodology [Simon Holloway, 1989]:

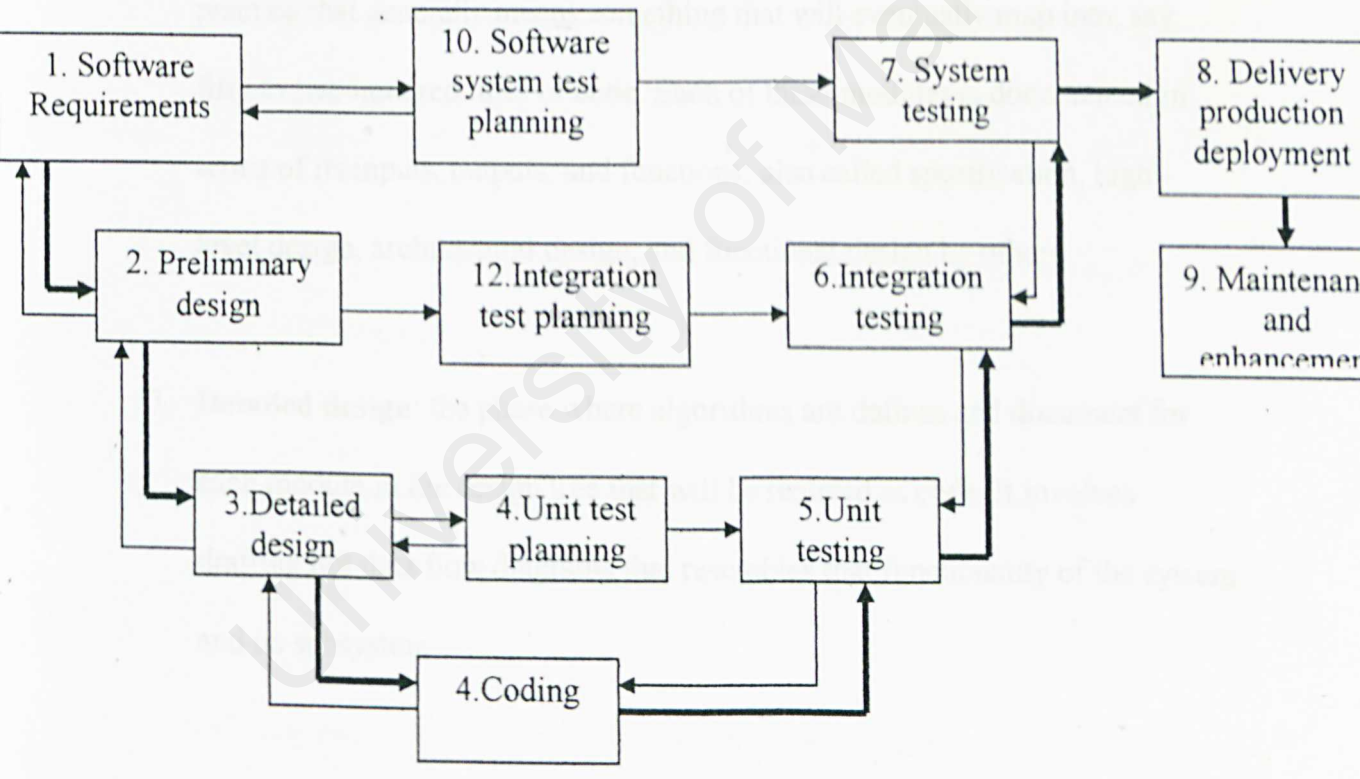
- Provides a standard framework that the developer does not have to reinvent the wheel for each project.
- Each method or tool in the methodology results in the successful completion of each development task.
- Reviews procedures are available to identify any errors, inconsistencies and discrepancies during development.
- Increase the system quality by forcing the developer to procedure flexible systems and adequate documentation.
- Provides better understanding of user needs and validation of user needs.
- Provides the management with tools to review project progress, and checklist to access tasks and deliverables.
- Improves communication among management, analyst, programmers, users and other stakeholders by providing a communication base.
- Facilitates planning and controlling the project.

A good methodology has the characteristic as below [Simon Holloway, 1989]:

- Easy to use for average analyst and programmers
- Covers all phases of system development
- Relevant to the type of application being developed (Transaction Processing System, Management Information System or others)
- Well quality documentation is available.
- Good vendor support in terms of training and consultancy.

Methodologies Review

Software development has been accepted as an engineering discipline because the process of software development is like other engineering processes. Hence, model of software development process was derived from other engineering discipline. This was well received by the industry as it offered a means of making the development process more visible and manageable. The Software Engineering Life Cycle (SDLC) has been chosen as the system process model. Figure 3.4 shows the Software Engineering Life Cycle (SDLC) this project.



Software Engineering Life Cycle

1. **Software requirements**: includes analyzing the software problem at hand and concludes with a complete specification of the desired external behavior of the software of the software system to built; also called functional description.

2. **Preliminary design**: decomposes the software system into its actual constituent (architectural) components and then iteratively decomposes those components into smaller and smaller subcomponents until the subcomponents located at the leaves of the resulting design tree are small enough so that we would expect a person to able to "get his or her arms around it" easily. In practice that generally means something that will eventually map into, say, fifty to two hundred lines of code. Each of these modules is documented in terms of its inputs, outputs, and functions; also called specification, high - level design, architectural design, and functional design by others.

3. **Detailed design**: the phase where algorithms are defines and document for each module in the design tree that will be realized as code. It involves drafting out data flow diagrams that resembles that functionality of the system and its subsystem.

4. Meanwhile, **coding phase** involved in transforming the algorithms defines during the previous phases into a computer understandable language. The program will be coded using selected programming languages and application development tools following the design specification.

5. The next phase is the **unit - testing phase**. The purpose of unit testing is to ensure that each module behave accordingly to its specification during program designed phase. It checks each coded module for the presence of bug.

6. **Integration testing**: interconnect sets of previously tested modules to ensure that the sets behave as well as they did as independently tested modules. Ideally each integrated set of modules should correspond to a component in the design tree defined during preliminary design. Thus, integration testing 's purpose is to ensure that each as - built component behaves according to its specification defined during preliminary design; also called string testing and computer software components (CSC) testing.

7. **System testing** is used to check the entire system to ensure that the system behaves according to the software requirement specification.

8. **Delivery, production, and development**: After final system testing, the software and its surrounding hardware become operational.

9. **Maintenance and enhancement**: The maintenance (continued detection and repair of bugs after development) and enhancement (addition of new capabilities) processes are actually a full development life cycle. The reason why it is a full – life cycle is simple: If a coding change is made, then the coding and three subsequent testing stages must be performed. If a design

change is made, then the design, coding, and three testing stages must be performed. If a requirement change has occurred, then all the stages must be performed.

10. **Software system test planning** assesses how the software system will be tested for conformity to the software requirements. It includes the development and documentation of test plans and procedures and might include the full – scale development of a test environment to test the actual system under test. The arrow in figure pointing from this stage back to the requirements stage represents the path taken in the event that a lack of verifiability is detected.

11. **Integration test planning**: generates and documents plans procedures to affect an orderly system integration. This might include developing specifications concerning the order of integration, test data to be used to test sets of components, and feedback to project management concerning relative priorities to apply during detailed design, coding, and unit testing in order to optimize the integration testing activity. This activity might also spawn a software development effort of its own to create scaffolding software, that is, software used to temporarily join two components during integration testing that would be removed before final system test.

12. **Unit test planning**: generates and documents plan and procedures to test each module independently and thoroughly.

Requirement analysis

A requirement is a feature of the system or a description of something that system must to do in order to achieve the objectives of the system. Basically system requirements fall into two major categories:

- **Functional requirements**
- **Non functional requirements**

Functional requirements

The functional requirements can be categorized into 3 sections, i.e. the main section modules, the authorized user section and the administrator section modules. A brief explanation of these modules is contained below.

Main section

- **Authentication and authorization modules**

  This module protects the system from unauthorized users. Authorized users will be provided with a user identification and password to access the system database.

- **Search information module**

  Users will use search engines to prompt the system to provide the information that they require.

- **Links module**

  This module enables users to surf in any website that are listed.

- **Sources module**

  This module enables users to refer any website that are related to this portal.

- **Quotable quotes**

  This module provides any random quotes.

- **Forum highlights module**

  This module enables users to give their comments to the topic that have been discussed.

- **News and activities module**

  This module provides any new activities that related to this portal.

- **Perplexed module**

  This module provides any hot topic or problems that related to this portal.

- **Top 10 Jobs module**

  This module enables users to know the top job from time to time.

- **Study tips of the day**

  This module provides good tips for the students to be successful.

Authorized users section

- Academic counseling module
- Career counseling module
- Relationship counseling module
- Chat module
- Forum module
- Quiz module
- Change password module

  This module allows users to change password for security purpose. User has to key in old password to ensure that a valid user is making changes. If old password is key in wrongly, the system will abort the password changes.

Administrator section

This section only allows users with 'management' status to access it, there are five modules that will be stated as below.

- **Approve user registration module**

  This module enables administrator to approve users' registration.

- **Record maintenance**

  The module enables administrator to maintain the records of users and counsellors by providing edit and delete options.

Non – functional requirements

Non – functional requirements are essential definition of system properties and constraint under which a system must operate. Although the non – functional requirements are subjective, they are as important as the functional requirements.

- **Reliability**

  It is reliability if the application system, software and hardware do not cause unnecessary failure or downtime when they are used in a reasonable manner.

- **User – friendly**

  The application system requires having a user – friendly interface for ease of usage. Graphical User Interface (GUI) approach should apply for better visual effect to user. Effective error handling and validation will also help the user to navigate to the system with more confidence, error message should be display to the user indicate what is going wrong rather than for the user to guess what is happening.

- **Usability**

  The application system shall be easy to use. There should no be complex and unnecessary step to perform. They shall enhance and support rather than limit or restrict business processes.

- **Respond time**

  The respond time should be within a reasonable interval time in retrieving any data or information. Good application systems should have a shorter respond time.

- **Maintainability and Expandability**

  The architecture and design should be able to maintain and can be extended if necessary amendment is required in the future.

- **Security**

  The portal shall be able to prevent unauthorized users access to the system.

- **Manageability**

  The portal, hardware and software shall be capable of being managed and easy to operate.

- **Robustness**

  The portal shall be able to handle or at least avoid disaster in the face of unexpected data.

Hardware and Software Requirements

For development purpose

The hardware requirements are as follows:

- At least Pentium
- At least 64MB of RAM
- A hard disk for at least 1GB of storage
- Other standard computer peripherals

The software requirements are as follows:

- Microsoft Windows 2000/ XP
- MySQL database server
- Apache Web Server
- PHP Triad 2-1-1
- Macromedia Dreamweaver MX
- Adobe Photoshop
- Flash
- MS Access

For Running Purpose

The hardware requirements are as follows:

- At least Pentium
- At least 64MB of RAM
- A hard disk for at least 1GB of storage
- Other standard computer peripherals

The software requirements are as follows

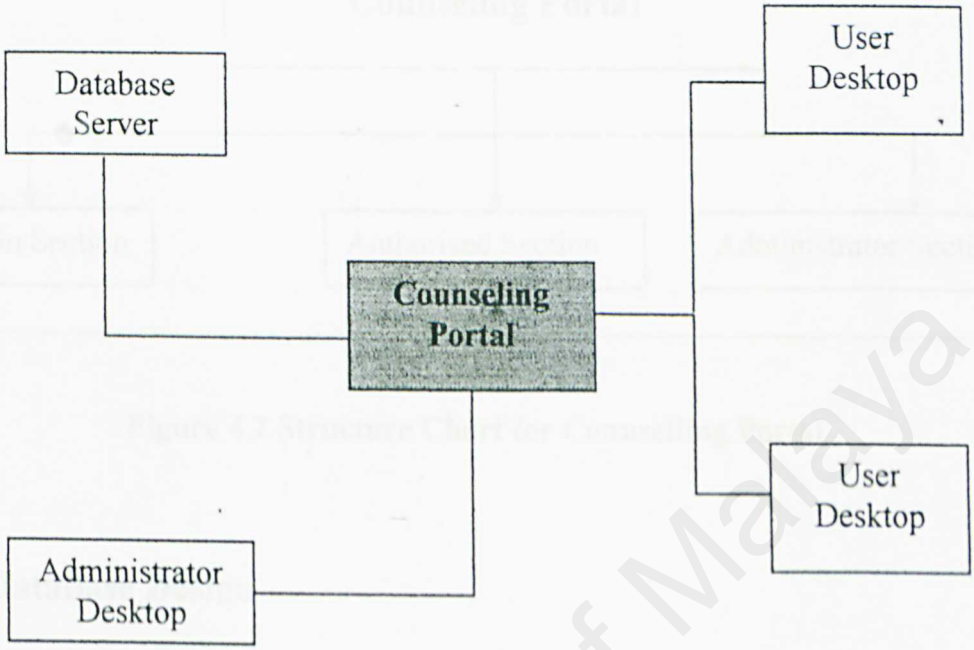
- Windows 98 and above
- Microsoft Internet Explorer

CHAPTER 4
System Design

System design

System design is a process where all the user requirements will be transformed into a real world application, which will be develop conceptually or logically. The purpose of system design is to select and plan system that meets the requirements needed to develop that system derived. System design is a very important stage to develop a system. A system design specification is needed to do system design. The design specification describes components or elements of a system and their appearance to the user. There are three stages in the system design process, they are architecture design, database design and user interface design.

Overview of System Architecture



Overview of Counselling Portal Architecture

4.2 Architecture Design

Architecture design is important to identify the subsystems that make up the system and their relationships. Each subsystem has its own function but it may relate to other subsystems to form a larger system.

The counseling portal will be made up of three sections, which are the Main Section, the Authorized User Section and the Administrator Section. Each section will consist of several subsystems that are built up smaller components or modules.

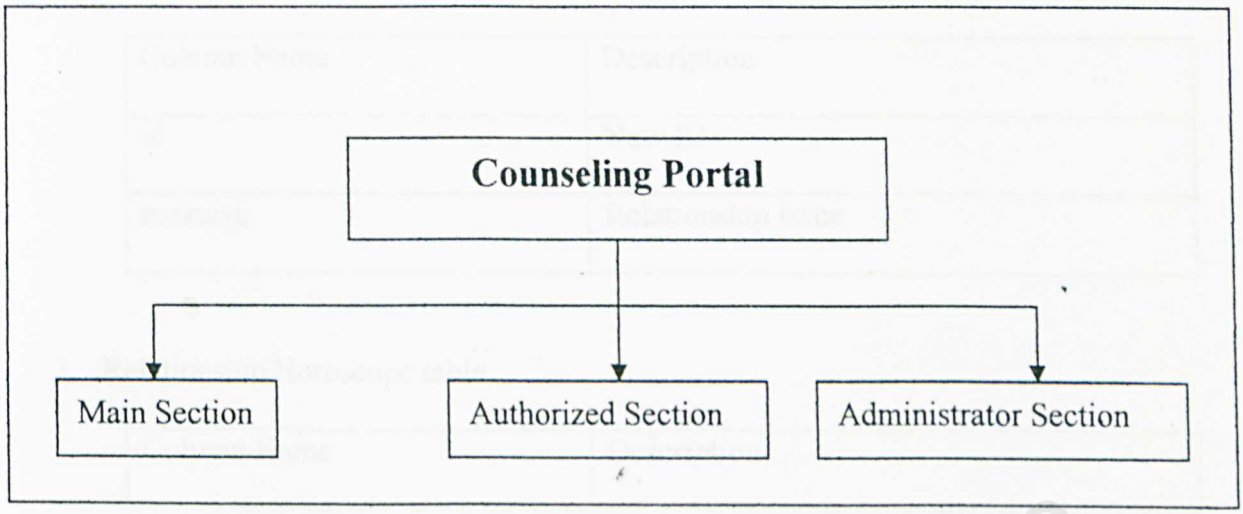


Figure 4.2 Structure Chart for Counselling Portal

4.3 Database Design

This involves identifying the business entities, their attributes and their relationships. Other business rules (validation rules, triggers) can also be added. Below is description of the database design for counseling portal system.

1. Perplexed

Column Name	Description
id	News ID
message	New message

2. Issue

Column Name	Description
id	New ID
message	Relationship issue

3. Relationship Horoscope table

Column Name	Description
message	Relationship horoscope

4. Quizzes Table

Column Name	Description
Quizzes	Quiz about relationship

5. Goodies table

Column Name	Description
tips1	Tips about relationship
tips2	Tips about relationship

6. Forum Table

Column Name	Description
bulletin_id	Bulletin ID
date	Date

time	Time
sender	Sender's name
sender_email	Sender's e – mail address
message	Message

7. Chat Table

Column Name	Description
id	ID
message	Message

4.4.2 Entity – Relationship Diagram

An entity – relationship diagram (E-R Diagram) is used to model the logical aspect of the system. It was introduced by Peter Chen in 1976 [Kroenke, 1998]. The E – R Diagram shows all the entities and relationship among them. An entity is something that can be identified in the user's work environment, something that the users want to track [Kroenke, 1998], an entity has the following characteristic [Don Yeates, Maura Shields and David Helmy, 1994]:

- It is interest to the organization.
- There is more than once instance of it.
- Each instance is uniquely identifiable.
- There is data to be held about it.

Each entity has attributes. An attribute is a data items belonging to the entity [Data Yeates, Maura Shields and David Helmy, 1994]. An identifier is an attribute that uniquely identifies one instances of the entity from others. An association between two entities is called the relationship between the two entities. A relationship can be one of three types, namely one-to-one (1:1), one-to-many (1:M), many-to-many (M: N) [Kendall & Kendall, 1999]. A relationship may also have attributes.

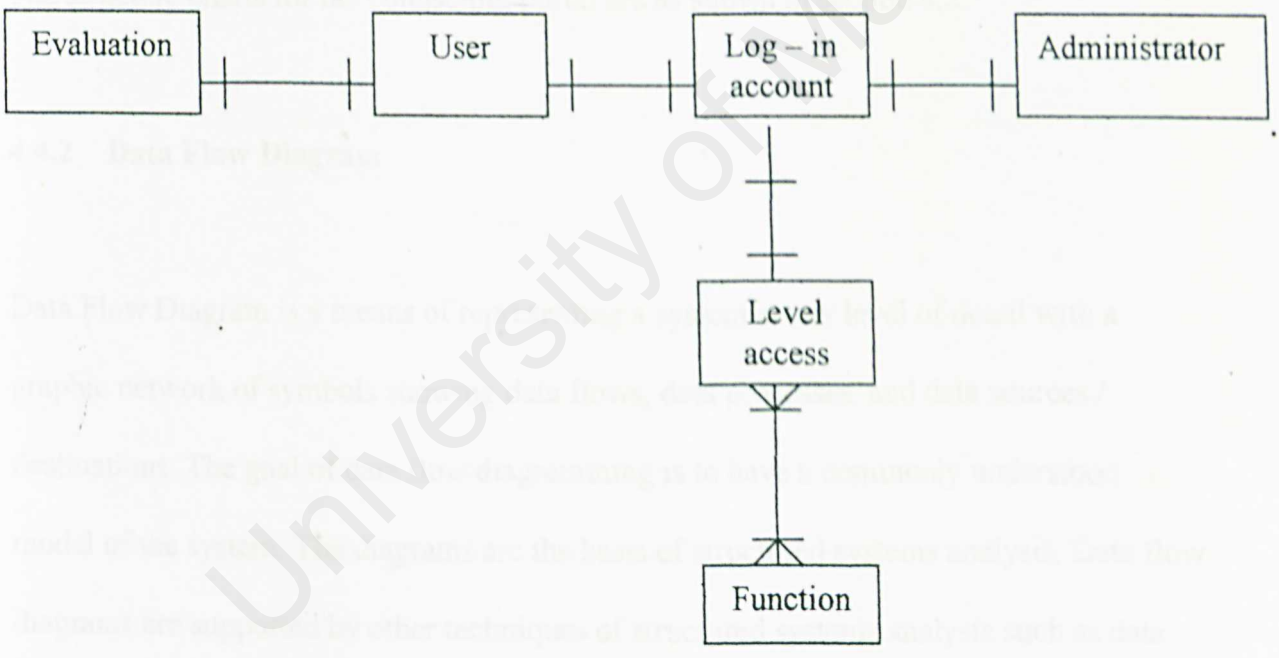


Figure 4.3: E – R Diagram for AskAkak.com

4.4 Process Design

There are several design methodologies for the process design. Counseling portal (relationship) is design based on data flow design method or structured design. Data flow oriented – design in earlier design concepts the stressed on modularity, top – down design and structured programming.

4.4.1 Structure Chart

The structure charts for the counseling portal are as shown in section 4.2.

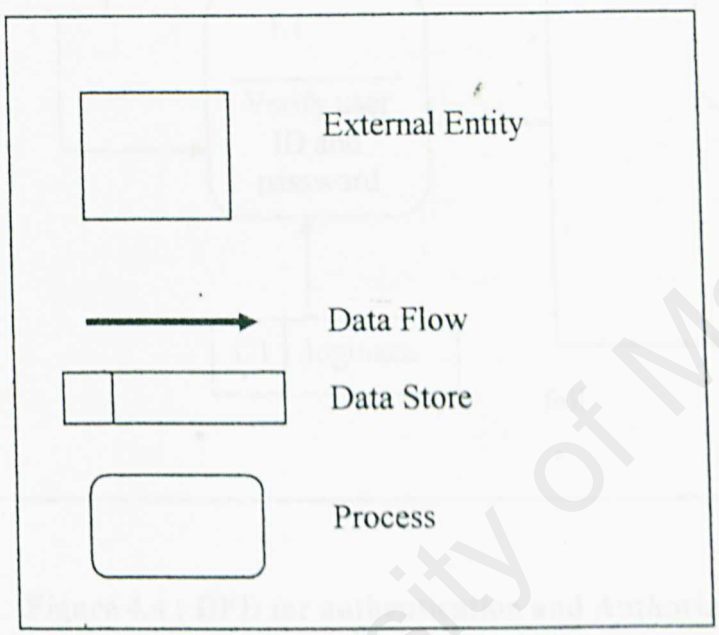
4.4.2 Data Flow Diagram

Data Flow Diagram is a means of representing a system at any level of detail with a graphic network of symbols showing data flows, data processes, and data sources / destinations. The goal of data flow diagramming is to have a commonly understood model of the system. The diagrams are the basis of structured systems analysis. Data flow diagrams are supported by other techniques of structured systems analysis such as data structure diagrams, data dictionaries and procedure-representing techniques such as decision tables, decision trees and structured English.

The purpose of data flow diagrams is to provide a semantic bridge between users and system developers. The diagrams are graphical, eliminating thousands of words; logical

representations, modeling WHAT a system does, rather than physical models showing HOW it does it; hierarchical, showing systems at any level of detail; and jargonless, allowing user understanding and reviewing.

Data Flowing Diagrams are composed of the four basic symbols shown below:



The External Entity symbol represents sources of data to the system or destinations of data from the system. The Data Flow symbol represents movement of data. The Data Store symbol represents data that is not moving. The Process symbol represents an activity that transforms or manipulates the data (combines, reorders, converts, etc).

For the counseling portal, the different type of users divides the module. There are global modules, which are accessible to all users, authorized users modules, and administrator modules.

The following figures illustrate the DFD for the modules in the counseling portal:

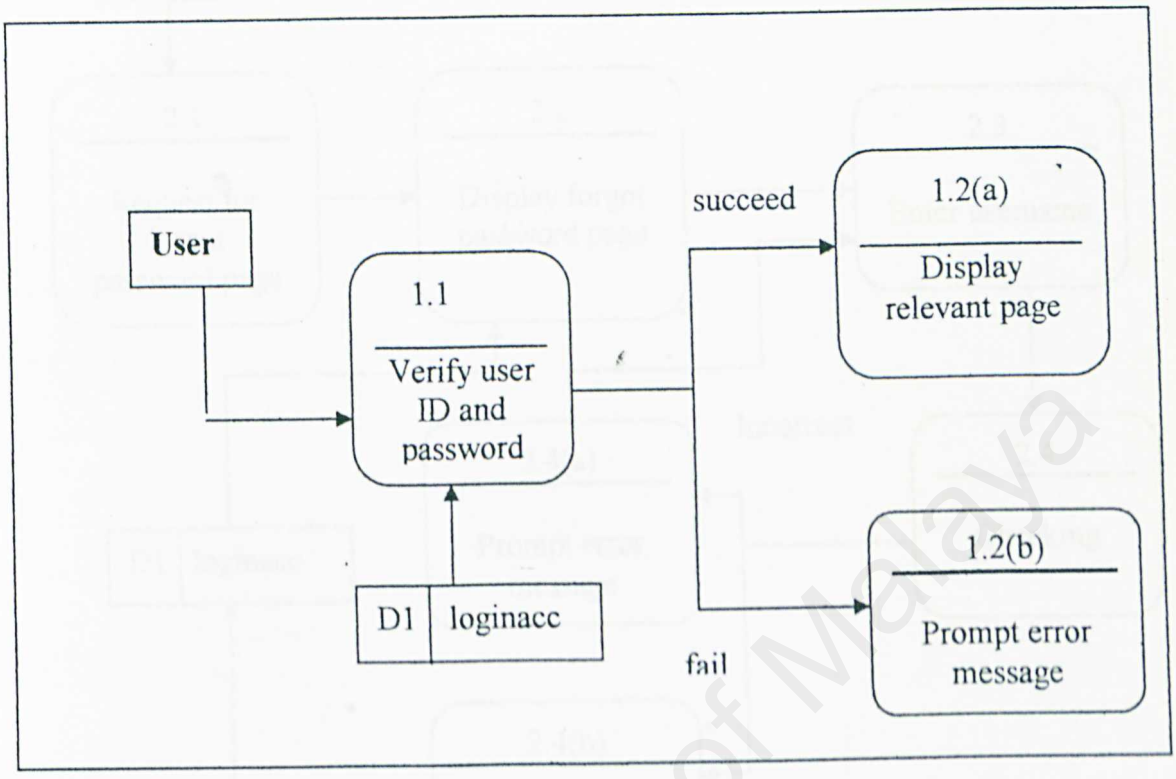


Figure 4.4 : DFD for authentication and Authorization Module

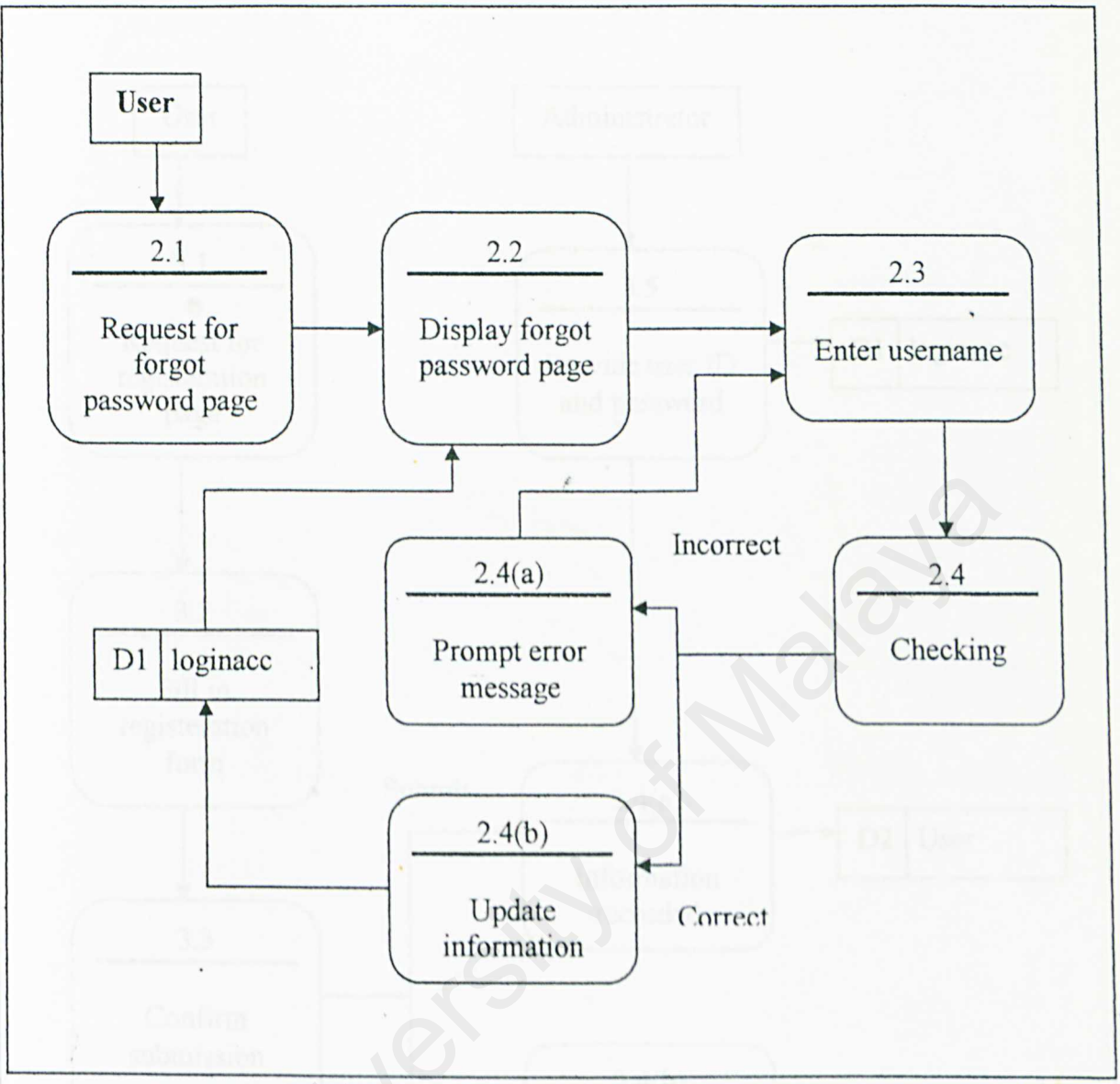


Figure 4.5: DFD for Forgot password Module

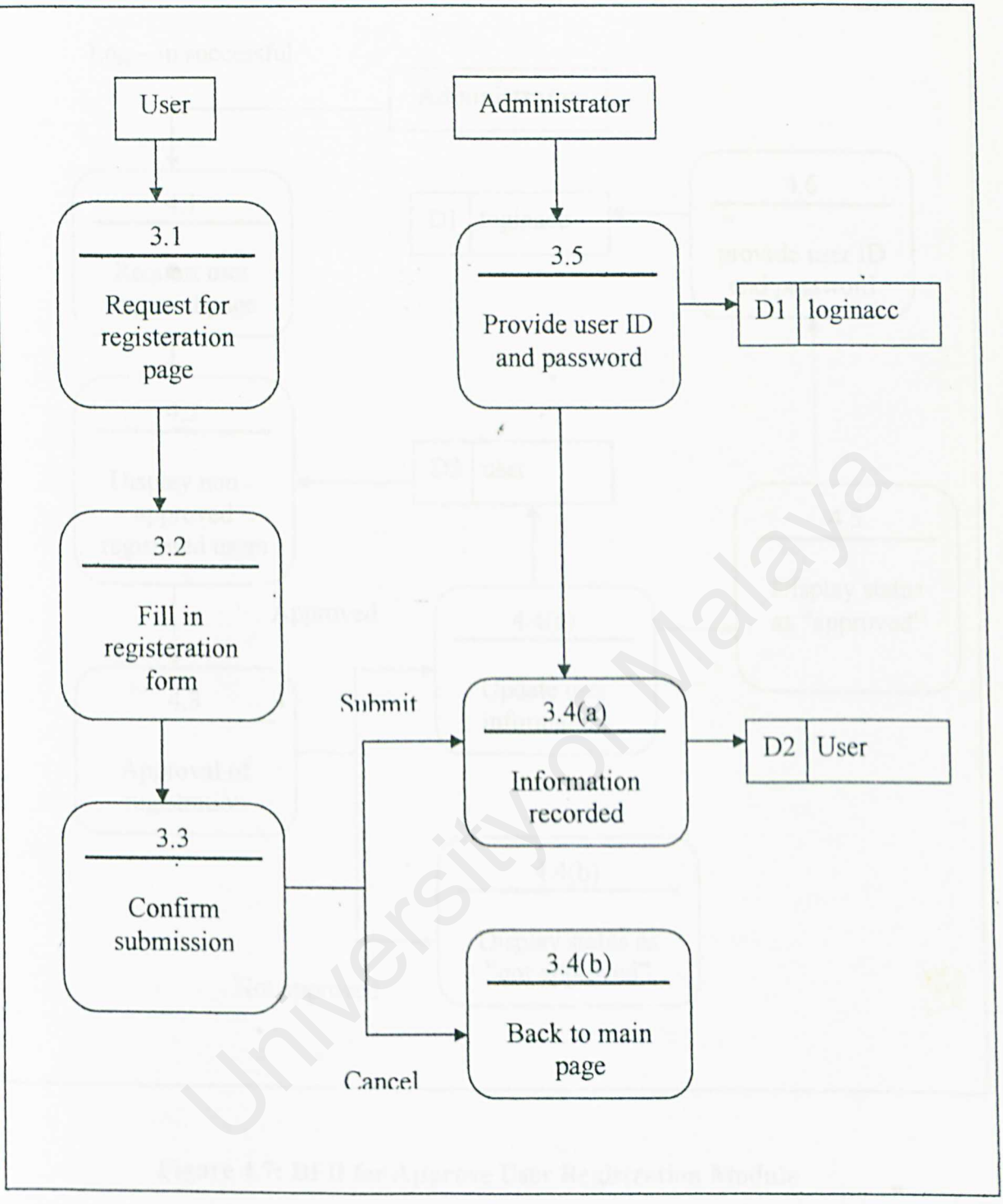


Figure 4.6: DFD for User Registration Module

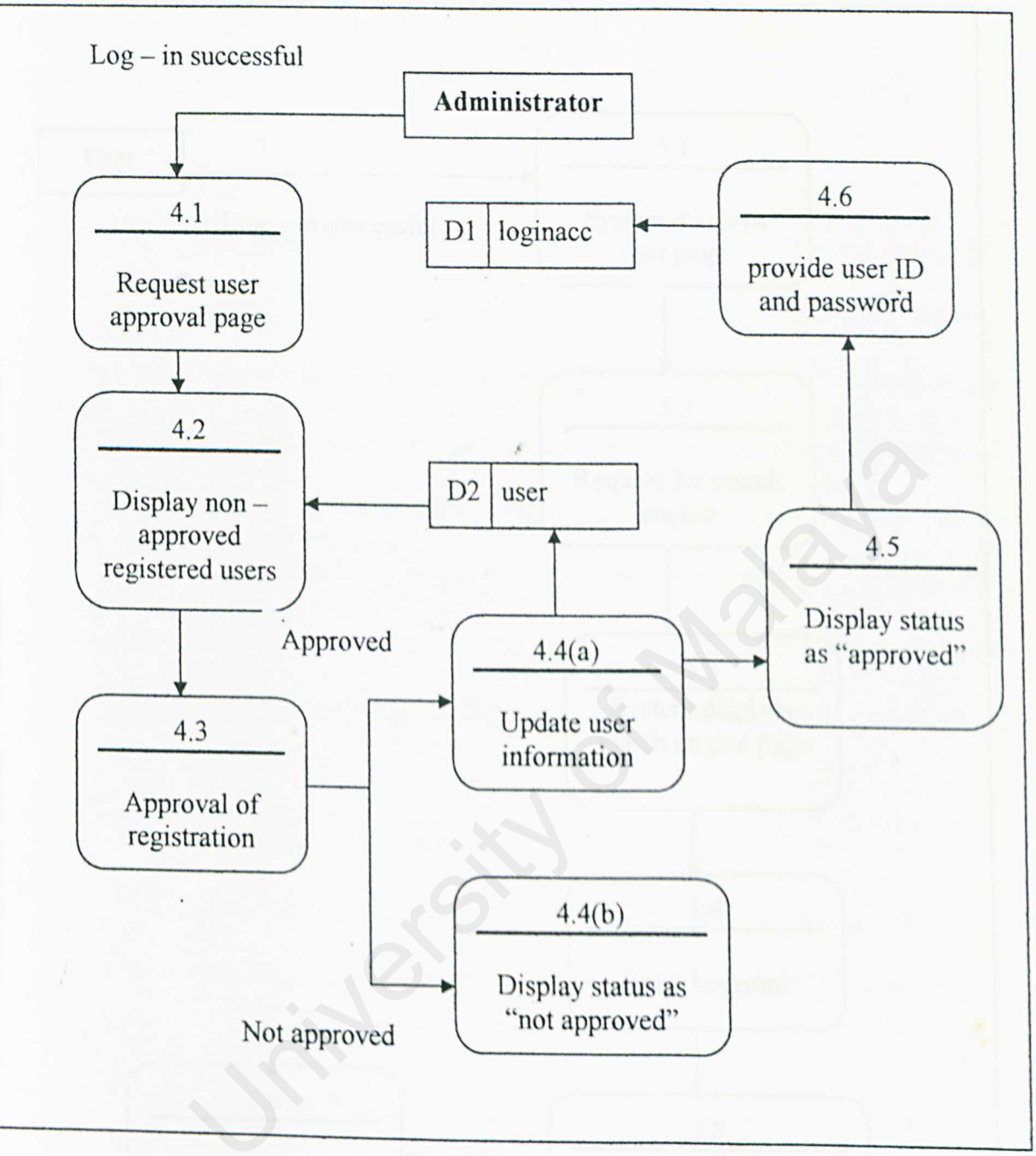


Figure 4.7: DFD for Approve User Registration Module

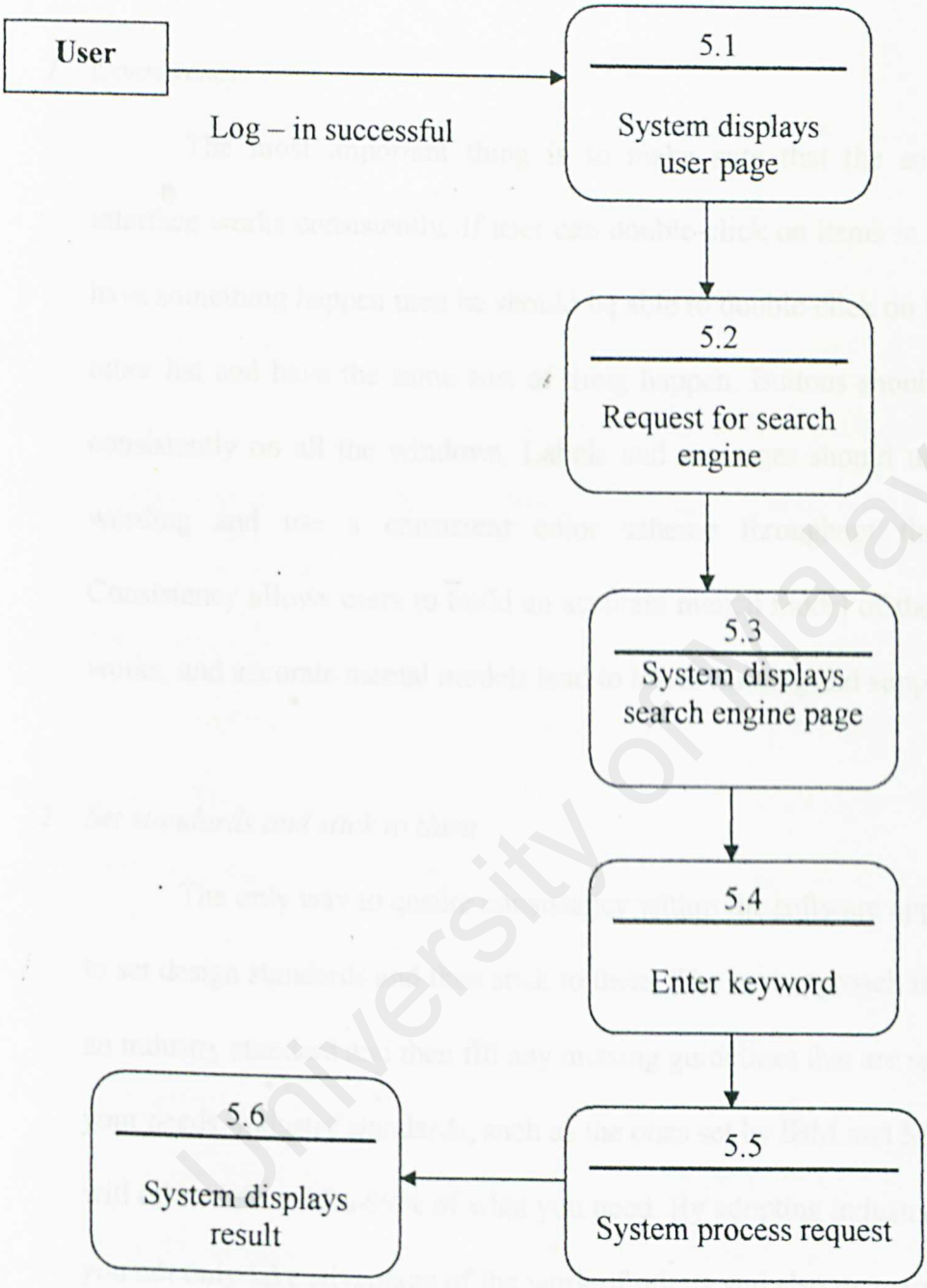


Figure 4.8: DFD for Search Engine Module

User Interface Design Tips and Techniques

1. *Consistency*

The most important thing is to make sure that the software user interface works consistently. If user can double-click on items in one list and have something happen then he should be able to double-click on items in any other list and have the same sort of thing happen. Buttons should be placed consistently on all the windows. Labels and messages should use the same wording and use a consistent color scheme throughout the software. Consistency allows users to build an accurate mental model of the way that it works, and accurate mental models lead to lower training and support costs.

2. *Set standards and stick to them*

The only way to ensure consistency within the software application is to set design standards and then stick to them. The best approach is to adopt an industry standard and then fill any missing guidelines that are specific to your needs. Industry standards, such as the ones set by IBM and Microsoft, will often define 95%-99% of what you need. By adopting industry standards you not only take advantage of the work of others you also increase the chance that your application will look and feel like other applications that your users purchase or have built. User interface design standards should be set during the Define Infrastructure Stage.

3. *Explain the rules*

Users need to know how to work with the application that is built for them. When an application works consistently it means the rules will need to be explain only once. This is a lot easier than explaining in detail exactly how to use each and every feature in an application step by step.

4. *Support both novices and experts*

This is to make sure that casual users can understand how the application works and does not boredom the expert users.

5. *Navigation between screens is important*

When the flow between screens matches the flow of the work that the user is trying to accomplish, then your application will make sense to your users. Interface-flow diagrams can be used during the Model Stage to model the flow between screens.

6. *Navigation within a screen is important*

In Western societies people read left to right and top to bottom. Because people are used to this should you design screens that are also organized left to right and top to bottom. You want to organize navigation between widgets on your screen in a manner that users will find familiar to them and related to the software application.

7. *Word messages and labels appropriately*

The text that will be displayed on the screens is a primary source of information for users. Using full words and sentences, as opposed to abbreviations and codes makes text easier to understand. Messages should be worded positively, imply that the user is in control, and provide insight into how to use the application properly. Furthermore, messages should be worded consistently and displayed in a consistent place on the screen.

8. *Understand the software widgets*

Software developer should use the right widget for the right task, helping to increase the consistency in the software application and probably making it easier to build the application in the first place. The only way to learn how to use widgets properly is to read and understand the user interface standards and guidelines that are available.

9. *Look at other applications with a grain of salt*

It is always a good idea to look at the work of others to get ideas, until you know how to distinguish between good user-interface design and bad user-interface design. Too many developers make the mistake of imitating the user interface of another application that was poorly designed.

10. Use color appropriately

Color should be used sparingly in the software applications, and to use it a secondary indicator must also be used. The problem is that some of the users may be colorblind – if color is used to highlight something on a screen then it will need something else to make it stand out so that people will notice,it, such as display a symbol beside it. Colors combination must also be consistent in the application so that it will have a common look and feel throughout the application. Also, color generally does not port well between platforms – what looks good on one system may look poor on another system.

11. Follow the contrast rule

The best way to make sure that screens are still readable contrast rule must be follow: Use dark text on light backgrounds and light text on dark backgrounds. It is very easy to read blue text on a white background but very difficult to read blue text on a red background. The problem is that there is not enough contrast between blue and red to make it easy to read, whereas there is a lot of contrast between blue and white.

12. Use fonts appropriately

Old English fonts might look good on the covers of William Shakespeare's plays, but they are really hard to read on a screen. Use fonts that are easy to read, such as serif fonts like Times Roman. Furthermore, use

your fonts consistently and sparingly. A screen using two or three fonts effectively looks a lot better than a screen that uses five or six.

13. Gray things out, do not remove them

Should the button be removed or grayed out? Gray it out, never remove it. By graying things out when they shouldn't be used people can start building an accurate mental model as to how your application works. If a widget or menu are simply remove instead of graying it out then it is much more difficult for the users to build an accurate mental model because they only know what is currently available to them, and not what is not available. The old adage that out of sight is out of mind is directly applicable here.

14. Use non destructive default buttons

It is quite common to define a default button on every screen, the button that gets invoked if the user presses the Return/Enter key. The problem is that sometimes people will accidentally hit the Enter/Return key when they do not mean to, consequently invoking the default button. The default button shouldn't be something that is potentially destructive, such as delete or save.

15. Alignment of fields

When a screen has more than one editing field it should be organize in a way that is both visually appealing and efficient. The best way to do so is to left-justify edit fields, or in other words make the left-hand side of each edit

field line up in a straight line, one over the other. The corresponding labels should be right justified and placed immediately beside the field. This is a clean and efficient way to organize the fields on a screen.

16. Justify data appropriately

For columns of data it is common practice to right justify integers, decimal align floating-point numbers, and left justify strings.

17. Do not create busy screens

Crowded screens are difficult to understand and hence are difficult to use. Experimental results (Mayhew, 1992) show that the overall density of the screen should not exceed 40%, whereas local density within groupings shouldn't exceed 62%.

18. *Group things on the screen effectively*

Items that are logically connected should be grouped together on the screen to communicate that they are connected, whereas items that have nothing to do with each other should be separated.

19. Open windows in the center of the action

When the user double-clicks on an object to display its edit/detail screen then his or her attention is on that spot. Therefore it makes sense to open the window in that spot, not somewhere else.

20. Pop-up menus should not be the only source of functionality

Users cannot learn how to use the application if it hides major functionality. One of the most frustrating practices of developers is to misuse pop-up, also called context-sensitive, menus. Typically there is a way to use the mouse on the computer to display a hidden pop-up menu that provides access to functionality that is specific to the area of the screen that you are currently working in.

CHAPTER 5
System Implementation

System implementation

System implementation is a process of writing the programming codes that implement the design. This task can be daunting for several reasons. First, the designer may not have addressed all of the idiosyncrasies of the platform and programming environment. Second, designer must write the codes in a way that is understandable to others. Third, designer must also take advantages of the characteristic of the design's organization, the data structure and the programming language's construct while still creating code that is easily reusable. Lastly, designer need to very familiar with the organization's standards and procedures so that not only others can understand what they have written but also why they are written.

Coding

Coding is the process of translating the design specification into source codes that can be process by the computer. Translation of the design into codes will become easier if the standards and procedures are in place. Standards and procedures can help the developer to organize their thoughts, avoid mistakes and maintain correspondence between design and code components.

Coding Approach

Various procedures involve methods for documenting code with the purpose of to construct an easier coding style and easy to follow. In addition to that, standardize

documentation can clarify the functions performed by each section of the program. Thus, good documentation can help to locate faults without difficulty and make changes effortlessly. Likewise, modifications to code that result from changes in hardware or interface specifications are straightforward and the likelihood of error to occur is minimized.

It is essential to have a direct correspondence between program design and code components. Design characteristic such as low coupling, high cohesion and well defined interfaces should be integrated so that algorithms, functions, interfaces and data structure can be traced easily from design code and vice versa.

Programming involves an immense of creativity and the design is a guide to the function or purpose of each component. On the other hand, the programmers have extensive flexibility in implementing the design as codes.

Coding Samples

1) Below is a sample of using PHP in database connection

```
<?php

if (eregi("sql_layer.php",$PHP_SELF)) {
   Header("Location: ../index.php");
   die();
}

/* $dbtype = "MySQL"; */

/*
 * sql_connect($host, $user, $password, $db)
 * returns the connection ID
 */


class ResultSet {
        var $result;
```

```
        var $total_rows;
        var $fetched_rows;

        function set_result( $res ) {
                $this->result = $res;
        }

        function get_result() {
                return $this->result;
        }

        function set_total_rows( $rows ) {
                $this->total_rows = $rows;
        }

        function get_total_rows() {
                return $this->total_rows;
        }

        function set_fetched_rows( $rows ) {
                $this->fetched_rows = $rows;
        }

        function get_fetched_rows() {
                return $this->fetched_rows;
        }

        function increment_fetched_rows() {
                $this->fetched_rows = $this->fetched_rows + 1;
        }
}


function sql_connect($host, $user, $password, $db)
{
global $dbtype;
switch ($dbtype) {

  case "MySQL":
    $dbi=@mysql_connect($host, $user, $password);
    mysql_select_db($db);
    return $dbi;
  break;;

  default:
  break;;
  }

}

function sql_logout($id)
{
global $dbtype;
switch ($dbtype) {
```

```
    case "MySQL":
      $dbi=@mysql_close($id);
      return $dbi;
    break;;

    default:
    break;;
    }
}


/*
 * sql_query($query, $id)
 * executes an SQL statement, returns a result identifier
 */

function sql_query($query, $id)
{

global $dbtype;
global $sql_debug;
$sql_debug = 0;
if($sql_debug) echo "SQL query: ".str_replace(",",", ",$query)."<BR>";
switch ($dbtype) {

    case "MySQL":
      $res=@mysql_query($query, $id);
      return $res;
    break;;

    default:
    break;;

    }
}

/*
 * sql_num_rows($res)
 * given a result identifier, returns the number of affected rows
 */

function sql_num_rows($res)
{
global $dbtype;
switch ($dbtype) {

    case "MySQL":
      $rows=mysql_num_rows($res);
      return $rows;
    break;;

    default:
    break;;
    }
```

```
}

/*
 * sql_fetch_row(&$res,$row)
 * given a result identifier, returns an array with the resulting row
 * Needs also a row number for compatibility with PostgreSQL
 */

function sql_fetch_row(&$res, $nr)
{
global $dbtype;
switch ($dbtype) {

   case "MySQL":
      $row = mysql_fetch_row($res);
      return $row;
   break;;

   default:
   break;;
   }
}

/*
 * sql_fetch_array($res,$row)
 * given a result identifier, returns an associative array
 * with the resulting row using field names as keys.
 * Needs also a row number for compatibility with PostgreSQL.
 */

function sql_fetch_array(&$res, $nr)
{
global $dbtype;
switch ($dbtype)
   {
   case "MySQL":
      $row = array();
      $row = mysql_fetch_array($res);
      return $row;
   break;;

   }
}

function sql_fetch_object(&$res, $nr)
{
global $dbtype;
switch ($dbtype)
   {
   case "MySQL":
      $row = mysql_fetch_object($res);
          if($row) return $row;
          else return false;
   break;;
```

```
    }
}

/*** Function Free Result for function free the memory ***/
function sql_free_result($res) {
global $dbtype;
switch ($dbtype) {

    case "MySQL":
        $row = mysql_free_result($res);
        return $row;
    break;;

            }
}

?>
```

2) Below is a sample of using PHP in *Search* module

```
<?php

if (!eregi("modules.php", $PHP_SELF)) {
    die ("You can't access this file directly...");
}

require_once("mainfile.php");
$module_name = basename(dirname(__FILE__));
get_lang($module_name);

if ($multilingual == 1) {
    $queryalang = "AND (s.alanguage='$currentlang' OR s.alanguage='')"; /* stories */
    $queryrlang = "AND rlanguage='$currentlang' "; /* reviews */
    $queryslang = "AND slanguage='$currentlang' "; /* sections */
} else {
    $queryalang = "";
    $queryrlang = "";
    $queryslang = "";
}

switch($op) {

        case "comments":
                break;

        default:
                $offset=10;
                if (!isset($min)) $min=0;
                if (!isset($max)) $max=$min+$offset;
                $query = stripslashes($query);
                        $pagetitle = "- "._SEARCH."";
                include("header.php");
```

```
                if ($topic>0) {
                  $result = sql_query("select topicimage, topictext from ".$prefix."_topics where
topicid=$topic", $dbi);
                  list($topicimage, $topictext) = sql_fetch_row($result, $dbi);
                } else {
                  $topictext = ""._ALLTOPICS."";
                  $topicimage = "AllTopics.gif";
                }
                OpenTable();
                if ($type == "users") {
                  echo "<center><font
class=\"title\"><b>"._SEARCHUSERS."</b></font></center><br>";
                } elseif ($type == "sections") {
                  echo "<center><font
class=\"title\"><b>"._SEARCHSECTIONS."</b></font></center><br>";
                } elseif ($type == "reviews") {
                  echo "<center><font
class=\"title\"><b>"._SEARCHREVIEWS."</b></font></center><br>";
                } else {
                  echo "<center><font class=\"title\"><b>"._SEARCHIN."
$topictext</b></font></center><br>";
                }
                echo "<table width=\"100%\" border=\"0\"><TR><TD>";
                if (($type == "users") OR ($type == "sections") OR ($type == "reviews")) {
                  echo "<img src=\"images/topics//info.gif\" align=\"right\" border=\"0\" alt=\"\">";
        } else {
                  echo "<img src=\"images/topics/$topicimage\" align=\"right\" border=\"0\"
alt=\"$topictext\">";
                }
                echo "<form action=\"modules.php?name=Search\" method=\"POST\">"
        ."<input size=\"25\" type=\"text\" name=\"query\" value=\"$query\">  "
                ."<input type=\"submit\" value=\""._SEARCH."\"><br><br>"
        ."<!-- Topic Selection -->";
                $toplist = sql_query("select topicid, topictext from ".$prefix."_topics order by topictext",
$dbi);
                echo "<select name=\"topic\">";
        echo "<option value=\"\">"._ALLTOPICS."</option>\n";
        while(list($topicid, $topics) = sql_fetch_row($toplist, $dbi)) {
            if ($topicid==$topic) { $sel = "selected "; }
            echo "<option $sel value=\"$topicid\">$topics</option>\n";
                    $sel = "";
        }
                echo "</select>";
                /* Category Selection */
                echo " <select name=\"category\">";
        echo "<option value=\"0\">"._ARTICLES."</option>\n";
                $catlist = sql_query("select catid, title from ".$prefix."_stories_cat order by title", $dbi);
        while(list($catid, $title) = sql_fetch_row($catlist, $dbi)) {
            if ($catid==$category) { $sel = "selected "; }
            echo "<option $sel value=\"$catid\">$title</option>\n";
                    $sel = "";
        }
                echo "</select>";
                /* Authors Selection */
        $thing = sql_query("select aid from ".$prefix."_authors order by aid", $dbi);
```

```
                echo " <select name=\"author\">";
        echo "<option value=\"\">"._ALLAUTHORS."</option>\n";
        while(list($authors) = sql_fetch_row($thing, $dbi)) {
            if ($authors==$author) { $sel = "selected "; }
                        echo "<option value=\"$authors\">$authors</option>\n";
                        $sel = "";
        }
        echo "</select>";
        /* Date Selection */
        ?>
             <select name="days">
          <option <?php echo $days == 0 ? "selected " : ""; ?> value="0"><?php echo _ALL
?></option>
          <option <?php echo $days == 7 ? "selected " : ""; ?> value="7">1 <?php echo _WEEK
?></option>
          <option <?php echo $days == 14 ? "selected " : ""; ?> value="14">2 <?php echo _WEEKS
?></option>
          <option <?php echo $days == 30 ? "selected " : ""; ?> value="30">1 <?php echo _MONTH
?></option>
                        <option <?php echo $days == 60 ? "selected " : ""; ?> value="60">2 <?php echo
_MONTHS ?></option>
          <option <?php echo $days == 90 ? "selected " : ""; ?> value="90">3 <?php echo
_MONTHS ?></option>
        </select><br>
            <?php
            if (($type == "stories") OR ($type == "")) {
              $sel1 = "checked";
            } elseif ($type == "comments") {
              $sel2 = "checked";
            } elseif ($type == "sections") {
              $sel3 = "checked";
            } elseif ($type == "users") {
              $sel4 = "checked";
            } elseif ($type == "reviews") {
              $sel5 = "checked";
            }

            $num_sec = sql_num_rows(sql_query("select * from".$prefix."_sections", $dbi), $dbi);
            $num_rev = sql_num_rows(sql_query("select * from ".$prefix."_reviews", $dbi), $dbi);

            echo ""._SEARCHON."";
            echo "<input type=\"radio\" name=\"type\" value=\"stories\" $sel1> "._SSTORIES."";
            echo "<input type=\"radio\" name=\"type\" value=\"comments\" $sel2>
"._SCOMMENTS."";
            if ($num_sec > 0) {
              echo "<input type=\"radio\" name=\"type\" value=\"sections\" $sel3>
"._SSECTIONS."";
            }
            echo "<input type=\"radio\" name=\"type\" value=\"users\" $sel4> "._SUSERS."";
            if ($num_rev > 0) {
              echo "<input type=\"radio\" name=\"type\" value=\"reviews\" $sel5> "._REVIEWS."";
      }
            echo "</form></td></tr></table>";
      $query = addslashes($query);
      if ($type=="stories" OR !$type) {
```

```
                if ($category > 0) {
                   $categ = "AND catid=$category ";
                } elseif ($category == 0) {
                   $categ = "";
                }
        $q = "select s.sid, s.aid, s.informant, s.title, s.time, s.hometext, s.bodytext, a.url, s.comments,
s.topic from ".$prefix."_stories s, ".$prefix."_authors a where s.aid=a.aid $queryalang $categ";
        if (isset($query)) $q .= "AND (s.title LIKE '%$query%' OR s.hometext LIKE '%$query%' OR
s.bodytext LIKE '%$query%' OR s.notes LIKE '%$query%') ";
        if ($author != "") $q .= "AND s.aid='$author' ";
        if ($topic != "") $q .= "AND s.topic='$topic' ";
        if ($days != "" && $days!=0) $q .= "AND TO_DAYS(NOW()) - TO_DAYS(time) <= $days ";
        $q .= " ORDER BY s.time DESC LIMIT $min,$offset";
                $t = $topic;
        $result = sql_query($q, $dbi);
        $nrows = sql_num_rows($result, $dbi);
        $x=0;
                echo "<br><hr noshade
size=\"1\"><center><b>"._SEARCHRESULTS."</b></center><br><br>";
        echo "<table width=\"99%\" cellspacing=\"0\" cellpadding=\"0\" border=\"0\">\n";
                if ($nrows>0) {
            while(list($sid, $aid, $informant, $title, $time, $hometext, $bodytext, $url, $comments,
$topic) = sql_fetch_row($result, $dbi)) {

                        $result2 = sql_query("select topictext from ".$prefix."_topics where
topicid=$topic", $dbi);
                        list($topictext) = sql_fetch_row($result2, $dbi);

                            $furl = "modules.php?name=News&file=article&sid=$sid";
                  $datetime = formatTimestamp($time);
                            $query = stripslashes($query);
                            if ($informant == "") {
                               $informant = $anonymous;
                            } else {
                               $informant = "<a
href=\"modules.php?name=Your_Account&amp;op=userinfo&amp;uname=$informant\">$informant</a>"
;
                            }
                            if ($query != "") {
                               if (eregi("$query",$title)) {
                                    $a = 1;
                               }
                               $text = "$hometext$bodytext";
                               if (eregi("$query",$text)) {
                                    $a = 2;
                               }
                               if (eregi("$query",$text) AND eregi("$query",$title)) {
                                    $a = 3;
                               }
                               if ($a == 1) {
                                    $match = _MATCHTITLE;
                               } elseif ($a == 2) {
                                    $match = _MATCHTEXT;
                               } elseif ($a == 3) {
```

```
                                        $match = _MATCHBOTH;
                                    }
                                    if (!isset($a)) {
                                        $match = "";
                                    } else {
                                        $match = "$match<br>";
                                    }
                                }
                    printf("<tr><td><img src=\"images/links/urlgo.gif\" border=\"0\" alt=\"\"> <font
 class=\"option\"><a href=\"%s\"><b>%s</b></a></font><br><font
 class=\"content\">"._CONTRIBUTEDBY." $informant<br>"._POSTEDBY." <a
 href=\"%s\">%s</a>",$furl,$title,$url,$said,$informant);
                    echo " "._ON." $datetime<br>"
                                ."$match"
                                .""._TOPIC.": <a
 href=\"modules.php?name=Search&amp;query=&amp;topic=$topic\">$topictext</a> ";
                                if ($comments == 0) {
                                    echo "("._NOCOMMENTS.")";
                                } elseif ($comments == 1) {
                                    echo "($comments "._UCOMMENT.")";
                    } elseif ($comments >1) {
                                    echo "($comments "._UCOMMENTS.")";
                                }
                                if (is_admin($admin)) {
                                    echo " [ <a
 href=\"admin.php?op=EditStory&amp;sid=$sid\">"._EDIT."</a> | <a
 href=\"admin.php?op=RemoveStory&amp;sid=$sid\">"._DELETE."</a> ]";
                                }
                                echo "</font><br><br><br></td></tr>\n";
                                $x++;
            }

                echo "</table>";
                } else {
            echo "<tr><td><center><font
 class=\"option\"><b>"._NOMATCHES."</b></font></center><br><br>";
                        echo "</td></tr></table>";
        }

        $prev=$min-$offset;
        if ($prev>=0) {
            print "<br><br><center><a
 href=\"modules.php?name=Search&amp;author=$author&amp;topic=$t&amp;min=$prev&amp;query=$q
 uery&amp;type=$type&amp;category=$category\">";
            print "<b>$min "._PREVMATCHES."</b></a></center>";
        }

        $next=$min+$offset;
            if ($x>=9) {
            print "<br><br><center><a
 href=\"modules.php?name=Search&amp;author=$author&amp;topic=$t&amp;min=$max&amp;query=$qu
 ery&amp;type=$type&amp;category=$category\">";
            print "<b>"._NEXTMATCHES."</b></a></center>";
        }
```

```
} elseif ($type=="comments") {

$result = sql_query("select tid, sid, subject, date, name from ".$prefix."_comments where
(subject like '%$query%' OR comment like '%$query%') order by date DESC limit $min,$offset", $dbi);
$nrows = sql_num_rows($result, $dbi);
$x=0;
echo "<br><hr noshade
size=\"1\"><center><b>"._SEARCHRESULTS."</b></center><br><br>";
echo "<table width=\"99%\" cellspacing=\"0\" cellpadding=\"0\" border=\"0\">\n";
if ($nrows>0) {
while(list($tid, $sid, $subject, $date, $name) = sql_fetch_row($result, $dbi)) {
$res = sql_query("select title from ".$prefix."_stories where sid='$sid'", $dbi);
list($title) = sql_fetch_row($res, $dbi);
$reply = sql_num_rows(sql_query("select * from ".$prefix."_comments
where pid='$tid'", $dbi), $dbi);
$furl = "modules.php?name=News&amp;file=article&amp;thold=-
1&amp;mode=flat&amp;order=1&amp;sid=$sid#$tid";
if(!$name) {
$name = "$anonymous";
} else {
$name = "<a
href=\"modules.php?name=Your_Account&amp;op=userinfo&amp;uname=$name\">$name</a>";
}
$datetime = formatTimestamp($date);
echo "<tr><td><img src=\"images/links/urlgo.gif\" border=\"0\" alt=\"\"> <font
class=\"option\"><a href=\"$furl\"><b>$subject</b></a></font><font
class=\"content\"><br>"._POSTEDBY." $name"
." "._ON." $datetime<br>"
."".": $title<br>";
if ($reply == 1) {
echo "($reply "._SREPLY.")";
if (is_admin($admin)) {
echo " [ <a
href=\"admin.php?op=RemoveComment&amp;tid=$tid&amp;sid=$sid\">"._DELETE."</a> ]";
}
echo "<br><br><br></td></tr>\n";
} else {
echo "($reply "._SREPLIES.")";
if (is_admin($admin)) {
echo " [ <a
href=\"admin.php?op=RemoveComment&amp;tid=$tid&amp;sid=$sid\">"._DELETE."</a> ]";
}
echo "<br><br><br></td></tr>\n";
}
$x++;
}

echo "</table>";
} else {
echo "<tr><td><center><font
class=\"option\"><b>"._NOMATCHES."</b></font></center><br><br>";
echo "</td></tr></table>";
}
```

```
        $prev=$min-$offset;
        if ($prev>=0) {
            print "<br><br><center><a
href=\"modules.php?name=Search&amp;author=$author&amp;topic=$topic&amp;min=$prev&amp;query
=$query&amp;type=$type\">";
            print "<b>$min "._PREVMATCHES."</b></a></center>";
        }

        $next=$min+$offset;
            if ($x>=9) {
            print "<br><br><center><a
href=\"modules.php?name=Search&amp;author=$author&amp;topic=$topic&amp;min=$max&amp;query
=$query&amp;type=$type\">";
            print "<b>"._NEXTMATCHES."</b></a></center>";
        }

        } elseif ($type=="reviews") {

        $result = sql_query("select id, title, text, reviewer, score from ".$prefix."_reviews where (title
like '%$query%' OR text like '%$query%') $queryrlang order by date DESC limit $min,$offset", $dbi);
        $nrows  = sql_num_rows($result, $dbi);
        $x=0;
            echo "<br><hr noshade
size=\"1\"><center><b>"._SEARCHRESULTS."</b></center><br><br>";
            echo "<table width=\"99%\" cellspacing=\"0\" cellpadding=\"0\" border=\"0\">\n";
            if ($nrows>0) {
          while(list($id, $title, $text, $reviewer, $score) = sql_fetch_row($result, $dbi)) {
                $furl = "modules.php?name=Reviews&amp;op=showcontent&amp;id=$id";
                $pages = count(explode( "<!--pagebreak-->", $text ));
            echo "<tr><td><img src=\"images/links/urlgo.gif\" border=\"0\" alt=\"\"> <font
class=\"option\"><a href=\"$furl\"><b>$title</b></a></font><br>"
                ."<font class=\"content\">"._POSTEDBY." $reviewer<br>"
                ."".._REVIEWSCORE.": $score/10<br>";
                if ($pages == 1) {
                echo "($pages "._PAGE.")";
            } else {
                echo "($pages "._PAGES.")";
                }
                if (is_admin($admin)) {
                echo " [ <a
href=\"modules.php?name=Reviews&amp;op=mod_review&amp;id=$id\">"._EDIT."</a> | <a
href=\"modules.php?name=Reviews.php&amp;op=del_review&amp;id_del=$id\">"._DELETE."</a> ]";
                }
            print "<br><br><br></font></td></tr>\n";
            $x++;
          }
                echo "</table>";
            } else {
          echo "<tr><td><center><font
class=\"option\"><b>"._NOMATCHES."</b></font></center><br><br>";
                echo "</td></tr></table>";
        }

        $prev=$min-$offset;
        if ($prev>=0) {
```

```
            print "<br><br><center><a
href=\"modules.php?name=Search&amp;author=$author&amp;topic=$t&amp;min=$prev&amp;query=$q
uery&amp;type=$type\">";
            print "<b>$min "._PREVMATCHES."</b></a></center>";
        }

        $next=$min+$offset;
            if ($x>=9) {
            print "<br><br><center><a
href=\"modules.php?name=Search&amp;author=$author&amp;topic=$t&amp;min=$max&amp;query=$qu
ery&amp;type=$type\">";
            print "<b>"._NEXTMATCHES."</b></a></center>";
        }

    } elseif ($type=="sections") {

        $result = sql_query("select artid, secid, title, content from ".$prefix."_seccont where (title like
'%$query%' OR content like '%$query%') $queryslang order by artid DESC limit $min,$offset", $dbi);
        $nrows  = sql_num_rows($result, $dbi);
        $x=0;
            echo "<br><hr noshade
size=\"1\"><center><b>"._SEARCHRESULTS."</b></center><br><br>";
            echo "<table width=\"99%\" cellspacing=\"0\" cellpadding=\"0\" border=\"0\">\n";
            if ($nrows>0) {
            while(list($artid, $secid, $title, $content) = sql_fetch_row($result, $dbi)) {
                $pages = count(explode( "<!--pagebreak-->", $content ));
                $result2 = sql_query("select secname from ".$prefix."_sections where
secid='$secid'", $dbi);
                list($sectitle) = sql_fetch_row($result2, $dbi);
                $surl =
"modules.php?name=Sections&amp;op=listarticles&amp;secid=$secid";
                $furl =
"modules.php?name=Sections&amp;op=viewarticle&amp;artid=$artid";
            echo "<tr><td><img src=\"images/links/urlgo.gif\" border=\"0\" alt=\"\"> <font
class=\"option\"><a href=\"$furl\"><b>$title</b></a></font><font
class=\"content\"><br>"._INSECTION.": <a href=\"$surl\">$sectitle</a><br>";
                if ($pages == 1) {
                    echo "($pages "._PAGE.")";
            } else {
                    echo "($pages "._PAGES.")";
                }
                if (is_admin($admin)) {
                    echo " [ <a
href=\"admin.php?op=secartedit&amp;artid=$artid\">"._EDIT."</a> | <a
href=\"admin.php?op=secartdelete&amp;artid=$artid&amp;ok=0\">"._DELETE."</a> ]";
                }
                echo "</font><br><br><br></td></tr>\n";
                $x++;
            }

            echo "</table>";
            } else {
          echo "<tr><td><center><font
class=\"option\"><b>"._NOMATCHES."</b></font></center><br><br>";
                echo "</td></tr></table>";
```

```
        }

        $prev=$min-$offset;
        if ($prev>=0) {
            print "<br><br><center><a
href=\"modules.php?name=Search&amp;author=$author&amp;topic=$t&amp;min=$prev&amp;query=$q
uery&amp;type=$type\">";
            print "<b>$min "._PREVMATCHES."</b></a></center>";
        }

        $next=$min+$offset;
            if ($x>=9) {
            print "<br><br><center><a
href=\"modules.php?name=Search&amp;author=$author&amp;topic=$t&amp;min=$max&amp;query=$qu
ery&amp;type=$type\">";
            print "<b>"._NEXTMATCHES."</b></a></center>";
        }

      } elseif ($type=="users") {

        $result = sql_query("select uid, uname, name from ".$user_prefix."_users where (uname like
'%$query%' OR name like '%$query%' OR bio like '%$query%') order by uname ASC limit $min,$offset",
$dbi);
        $nrows  = sql_num_rows($result, $dbi);
        $x=0;
            echo "<br><hr noshade
size=\"1\"><center><b>"._SEARCHRESULTS."</b></center><br><br>";
            echo "<table width=\"99%\" cellspacing=\"0\" cellpadding=\"0\" border=\"0\">\n";
            if ($nrows>0) {
           while(list($uid, $uname, $name) = sql_fetch_row($result, $dbi)) {
                $furl =
"modules.php?name=Your_Account&amp;op=userinfo&amp;uname=$uname";
                if ($name=="") {
                    $name = ""._NONAME."";
                }
            echo "<tr><td><img src=\"images/links/urlgo.gif\" border=\"0\" alt=\"\"> <font
class=\"option\"><a href=\"$furl\"><b>$uname</b></a></font><font class=\"content\"> ($name)";
                if (is_admin($admin)) {
                    echo " [ <a
href=\"admin.php?chng_uid=$uid&amp;op=modifyUser\">"._EDIT."</a> | <a
href=\"admin.php?op=delUser&amp;chng_uid=$uid\">"._DELETE."</a> ]";
                }
                echo "</font></td></tr>\n";
            $x++;
           }

            echo "</table>";
            } else {
           echo "<tr><td><center><font
class=\"option\"><b>"._NOMATCHES."</b></font></center><br><br>";
                echo "</td></tr></table>";
        }

        $prev=$min-$offset;
        if ($prev>=0) {
```

```
                print "<br><br><center><a
href=\"modules.php?name=Search&amp;author=$author&amp;topic=$t&amp;min=$prev&amp;query=$q
uery&amp;type=$type\">";
                print "<b>$min "._PREVMATCHES."</b></a></center>";
            }

            $next=$min+$offset;
                if ($x>=9) {
                print "<br><br><center><a
href=\"modules.php?name=Search&amp;author=$author&amp;topic=$t&amp;min=$max&amp;query=$qu
ery&amp;type=$type\">";
                print "<b>"._NEXTMATCHES."</b></a></center>";
            }

            }
    CloseTable();
    include("footer.php");
    break;
}

?>
```

CHAPTER 6
System Testing

System Testing

Software errors and failures occur mainly because of inadequate or improper testing. Quality software however demands that software be tested carefully. Testing is not the first place where faults finding occurs because the requirements and design reviews helped to ferret out the problems early in the development. According to Alka Jarvis , there is a different between the four basic concepts related to software testing. There are :

1. Error detection - It involves identifying errors, inspection and walk through in the unit level.
2. Error removal - It involves debugging and other strategies for identifying where the error occurs in the code
3. Error tracking - It is important to find and correct the cause of the error, as it is to fix the error itself.
4. Regression testing - It is testing to see if the fix or rework to the code actually fixes the error, fixes it in one place and breaks it in another, or breaks the code in other places without actually fixing it at the point of the software where the fix was attempted.

The purpose of testing is to detect the presence of errors in AskAkak.com to maintain its quality. There are several testing principles that have been followed to ensure that AskAkak.com are well tested.

The principles adopted are as follows :

1. Test should be planned before testing begins.
2. All tests should be traceable to the requirements, which means that AskAkak.com must meet all the requirements of the customer.
3. Testing should begin in the small and progress towards testing in the large.

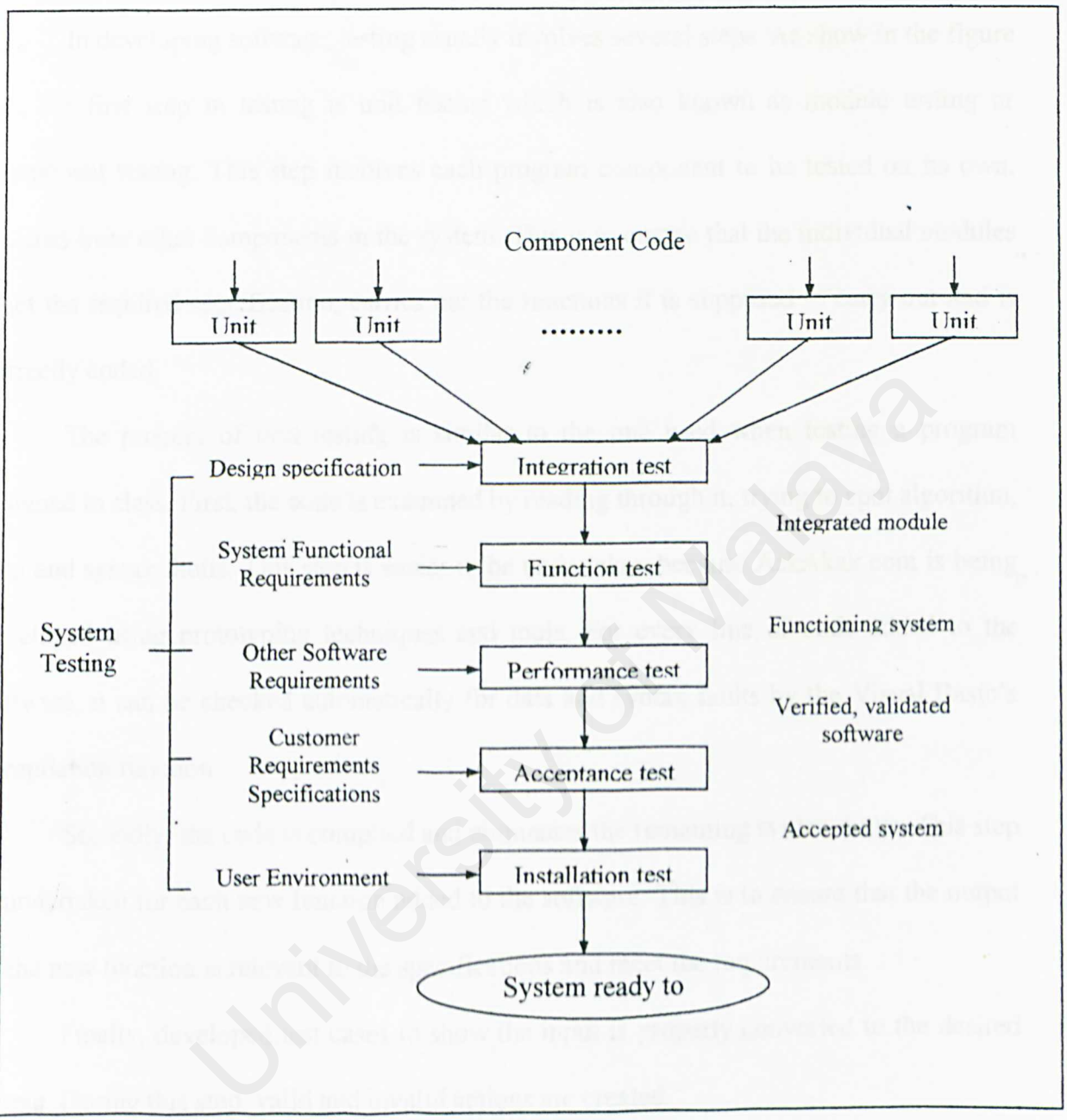


Figure 6-1 Levels of System Testing

6.1 Unit Testing

In developing software, testing usually involves several steps. As show in the figure 6-1, the first step in testing is unit testing which is also known as module testing or component testing. This step involves each program component to be tested on its own, isolated from other components in the system. This is to ensure that the individual modules meet the required specification, carries out the functions it is supposed to carry out and is correctly coded.

The process of unit testing is similar to the one used when testing a program assigned in class. First, the code is examined by reading through it, trying to spot algorithm, data and syntax faults. This step is easier to be undertaken because AskAkak.com is being developed using prototyping techniques and tools. For every line of code added to the software, it can be checked automatically for data and syntax faults by the Visual Basic's compilation function.

Secondly, the code is complied and eliminates the remaining syntax faults. This step is undertaken for each new function added to the software. This is to ensure that the output of the new function is relevant to the specifications and meet the requirements.

Finally, developed test cases to show the input is properly converted to the desired output. During this step, valid and invalid actions are created.

6.1 Integration Testing

When all the individual components or functions have satisfactory results which show that they are working correctly and meet the software objectives. Then all the

components are integrated or combined into a working system. This integration is planned and coordinated so that when a failure occurs, it can be solved immediately. The goal is to determine if the system or subsystem meets the system requirements and functions properly and to test interfaces among the modules.

The system is viewed as a hierarchy of components, where each component belongs to a layer of the design. It can begin from the top and work the way down as testing is done, work from the bottom up or use a combination of these two approaches.

There are four major approaches for merging components to test the larger system

1. Bottom up integration
2. Top down integration
3. Bing bang integration
4. Sandwich integration

Among these approaches, the Top down Integration is used in the ASkAkak.com integration testing where testing begins from the top and works the way down.

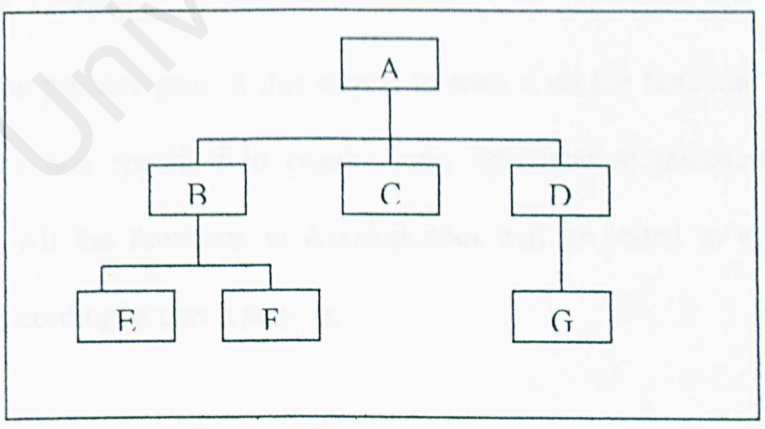


Figure 6-2 Example Component Hierarchies

6.3 System testing

Finally, system testing is performed. Testing the system is very different from unit and integration testing. The objective of unit and integration testing was to ensure that the code implemented was design properly. In system testing, the objective is to ensure that the system does what the user wants it to do. To perform this task, developer must work together with user. Because system testing takes place at a higher level, the testing focuses on behavior rather than function or functional structure and this cannot be tested through code audits based on pattern matching. The system testing result will show whether the entire system requirements, specification and objectives are achieved.

There are several steps in testing a system which are as follow :

1. Function testing
2. Performance testing
3. Acceptance testing
4. Installation testing

6.3.1 Function Testing

The primary goal of this step is to tests if all the functions required by the application and specified in requirements specification documents are working properly. All the functions in Askakak.com will be tested to ensure that it can operate according to user's actions.

6.3.2 Performance Testing

When all the system function work perfectly according to specifications the performance test is conducted. This testing step will compare the integrated components with the nonfunctional system requirements. These requirements, including security, accuracy, speed and reliability, constrain the way in which the system functions are performed. The S is tested to evaluate the followings :

1. Security precautions
2. Precisions accuracy in data, references, functions and process
3. Speed of data retrieval
4. System reliability and robustness
5. Response time to user's actions and error detection

When the system operates the way it was design, it is called a verified system. This verified system is the designers' interpretation of the requirements specification. Next, it is compared with the user's expectations by reviewing the requirements definition. Upon satisfaction, the system is now called a validated system, which verified that the requirements have been met.

6.3.3 Acceptance Testing

So far all the test have been done by the developer based on understanding of the system, its objectives and requirements. To ensure that the system meets the user's understanding of the requirements, which maybe different from the developer's point of

view, the system must be tested by the user them. This purpose of acceptance test is to confirm that the system is developed according to the user's requirements and it is ready for operational use. Normally, it is part of a formal handoff or release process.

During an acceptance test, end users of the system compare the system to its requirements. A few measures that can be adopted during this test are :

1. Execute the test procedures
2. Evaluate test result
3. Check for discrepancies
4. Accept or reject the system based on the acceptance test criteria established in the beginning of the project

6.4 Testing Result

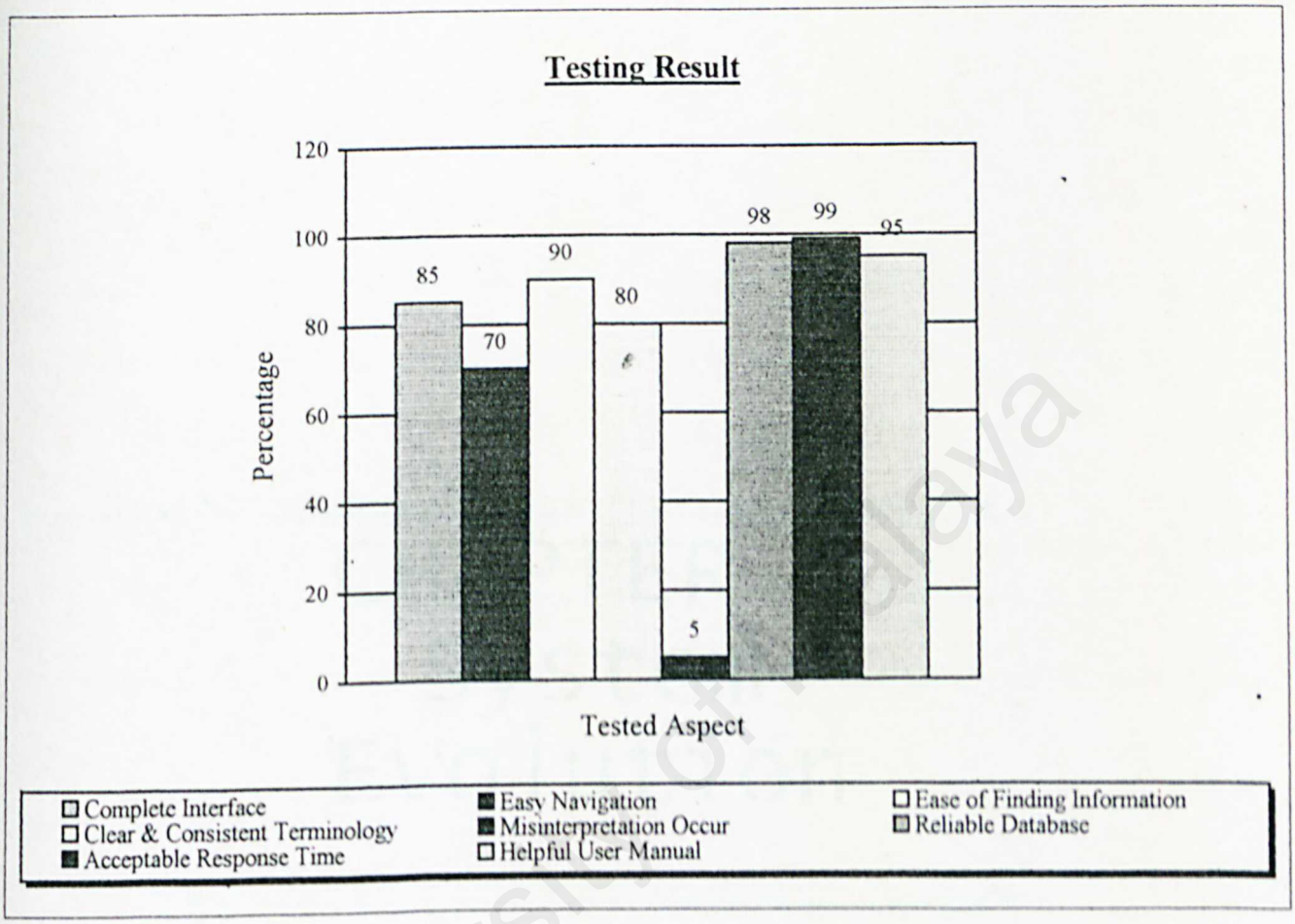


Figure 6-4 Testing Result Graph

CHAPTER 7
System Evolution

System Evaluation

Introduction

When all the previous task and steps have been undertaken and the results were satisfactory, the system is then evaluated to determine the system's strength, limitation, constraints and problems encountered during the system development process. It also highlights on the knowledge gained and identifies the steps taken in solving problems.

Problem Encountered and Solutions

Problems always occur during any development project and the development process of AskAkak.com has been no exception. Throughout the development of Askakak.com, many problems arise due to may reasons that will be stated below.

i) Lack of Experience in the Chosen Programming Language

Due to inexperience in PHP, the functions provided in PHP environment cannot be manipulated to the maximum extent. This is because all the previous semester subjects do not require knowledge in PHP. However, with the edge of Internet today, PHP and its environment can be learned through thousands of relevant example which are free and can be downloaded easily. Therefore, these advantages had miraculously helped to enhance knowledge in PHP and its environment.

ii) Determining the Functions of the System

In view of the fact that there was no prior experience in developing a new system, it was quite difficult to determine to which extent to define the scope of the system consequently it can be developed within the given time frame. In spite of this, this problem was overcome with the help from the lecturers, friends and analyzing existing relevant system.

iii) Time Constraint

Throughout the analysis and design phase in Semester 2 there was not sufficient time to study and create the best solution of the system interface design. Primarily, this was due to inexperience and inadequate knowledge in designing a design system. Therefore, when system implementation is undertaken, there were many changes made to the original interface design of the system. Nevertheless, with the tips and techniques in user interface design that was analyze in the system design this problem was overcome.

iv) Lack of Knowledge in Counselling Information

Even though there was insufficient knowledge in Counselling and but with numerous references on the Internet and in books, compiling the counseling was easier. On the other hand, since there was countless counseling information to choose from choosing which one to be used in the system was quite tricky. However, it was decided only four from the best and most accurate was chosen.

v) Lack of Skills in Creating Graphical Effects

Creating graphical effects was a big challenge in this project. Generally, this is due to lack of experience and interest. These effects consume a lot of time and try and error action. Nevertheless, the AskAkak.com have several graphical effects which are simple and does not consume a lot of time to load.

System Strength

All of the AskAkak.com software assets are discussed below :

1) High Usability

The AskAkak.com have a simple yet elegance user interface. This is because the AskAkak.com generally will be use most by students and adults. Since the AskAkak.com was developed based on HCI principle, the interface is intuitive where users can easily get accustomed to the system functions and style. On the whole, users won't face any problem using the AskAkak.com even though they are computer illiterate.

2) Transparent

The system is transparent as users do not need to know how the database performs its search, how the system is structured, etc. As an example, when user needs to search any information about counselling , they only have to type the word and click on the specific function and follow the instruction. They do not need to know how to retrieve information from the database.

3) *Users feedback*

The user can submit their comments and opinions to the developer of AskAkak.com via email in the feedback section. From the user's feedback, the developer will acknowledge whether the AskAkak is working perfectly in fulfilling user's need and request and changes can also be made to improve the AskAkak.com.

System Limitation

In spite of the strengths discussed in the later section, there are a number of limitations which the AskAkak.com cannot performed due to lack of research and time constrain throughout the development stage. These limitations are as follow :

1) The search function is only restricted to the words provided in the search database. Users cannot search beyond the counselling portal.
2) .The scope of the AskAkak.com only covers a small amount of information of Counselling which are academic, career and relationship counselling.
3) Enhancement of the graphical user interface should be done, for example by adding icons, with more stress on the concept of Human – Computer Interaction (HCI) in building GUIs to produce a system with high level of usability.

Conclusion

Conclusion

On the whole, this project has achieved the objectives and user requirements that have been determined in the system analysis process. It has also managed to be completed within the time given by the lecturer. Moreover, the user can communicate with the AskAkak.com developer via email to convey their suggestions and comments. Furthermore, the response time to display output based on user's request is toleratable. Since the design of the user interface are based on HCI and approved guidelines, the system should be able to use by anyone especially university students even though they are computer illiterate.

There was scores of knowledge and experience gained throughout this project. It does not cover only the technical and theoretical part of the development process but also enhance the knowledge on counselling specifically.

There are several new concepts have been learned during the development of this project such as the HCI concept. Learning the PHP and its environment was priceless knowledge and unforgettable experience. Furthermore, skills in using new software such as PHP Triad 2-1-1and Dreamweaver MX have been acquired and it is also a very significant experience.

Although programming skills are vital in any system development, high-quality practice on software engineering techniques must also be applied proficiently. This is to ensure that not only the system developed will meet all the user requirements but also a system with high quality standards. This project has provide the requisite opportunity to

apply all the techniques, paradigms, concepts and approaches learned from System Analysis and Design, Software Quality and Software Engineering courses.

Conversely, there is still space of improvement that could be done in the AskAkak.com as mentioned earlier.

In conclusion, this project was a very good opportunity for students to illustrate their skill in developing a system. This is an outstanding practice to guide and prepare students towards the working environment.

References

References

[1] Bruce, C. S. (1994) 'Research student's early experiences of the dissertation literature review' Studies in Higher Education, vol. 19, no. 2, pp. 217-229.

[2] Bruce, C. (1994) 'Supervising literature reviews', in Zuber-Skerritt, O. and Ryan, Y. (eds), Quality in postgraduate education, Kogan Page, London.

[3] Afolabi, M. (1992) 'The review of related literature in research' International journal of information and library research, vol. 4, no. 1, pp. 59-66.

[4] Bourner, T. (1996) 'The research process: four steps to success', in Greenfield, T. (ed), Research methods: guidance for postgraduates, Arnold, London.

[5] Cooper, H. M. (1988) 'The structure of knowledge synthesis' Knowledge in Society, vol. 1, pp. 104-126

[6] Bruce, C. S. (1990) 'Information skills coursework for postgraduate students: investigation and response at the Queensland University of Technology' Australian Academic & Research Libraries, vol. 21, no. 4, pp. 224-232.

[7] Dena Taylor (2000) 'The Literature Review: A Few Tips On Conducting It' URL: http://www.infocommons.utoronto.ca/writing/litrev.html

[8] Dr. P. Sellapan (2000) 'Software Engineering Management & Methods', Sejana Publishing, Malaysia

[9] F.O.S Inc. & Charles M. Carroll (1997) 'Visual Basic Information' URL: http://www.thebestweb.com/VBFAQS/faq.vbinfo.asp

[10] Scott Berkum (2000) 'The Art of UI Prototyping'
URL: http://msdn.microfoft.com/ui/

[11] Ayean Gulez (2000) 'Macromedia-Tool Information'
URL: http://www.wowwebdesign.com/tools/id_8

[12] October 2001 Techie Toy of the Month 'Swish 2.0'
URL: http://www.techietoyofthemonth.html

[13] Bret Holtz (1998) 'Microsoft Access, Visual Basic; Which to use?'
URL: http://w3.arizona.edu/~ccitinfo/newsletter,mayjune98/extendu.html

[14] Arch640 Development Team (1995) 'Photoshop home Page'
URL: http://www.rice.edu/computer/Tutorials/ravl/pshop//

[15] Adobe System Inc. (2001) 'Adobe Photoshop 6.0'
URL: http://www.pacific.adobe.com/products/photoshop/main.html

[16] International System Consultancy (2000) 'Home of ParsNegar'
URL : http://www.isc.com.au/

[17] Kandall Bass (1997) 'A Brief Guide to Introduction Multimedia and the Study of United States'
URL: http://www.georgetown.edu/crossroads/mltmedia.html#2

[18] 'Business Week Magazine', Feb 28 1994

[19] Cox Family (1996) 'Computer as Teacher and Tutors'
URL: http://www.concentric.net/~skiplac/computer.html#INTRODUCTION

APPENDIX A
User Manual

About This Manual

This manual will guide you on the following area:

- Hardware and Software Requirements
- Compatibilities
- Configuring the Counselling Portal
- Tour Guide through AskAkak.Com Counselling Portal

Hardware & Software Requirements

Hardware Requirements

Listed below are the minimum hardware requirements for running the AskAkak.Com Counselling Portal:

- Pentium 233 MHz or above (Runs well in Celeron and AMD Processor also)
- 16MB RAM
- Other basic requirements for a desktop computer

Software Requirements

AskAkak.Com Counselling Portal runs well in:

- Windows 98 Second Edition (SE) Operating System
- Internet Explorer 4 and above Web Browser
- Apache HTTP Web Server

Compatibilities

AskAkak.Com Counselling Portal has been tested in various Windows OS platforms and different web browsers. Below is the result of the tests conducted:

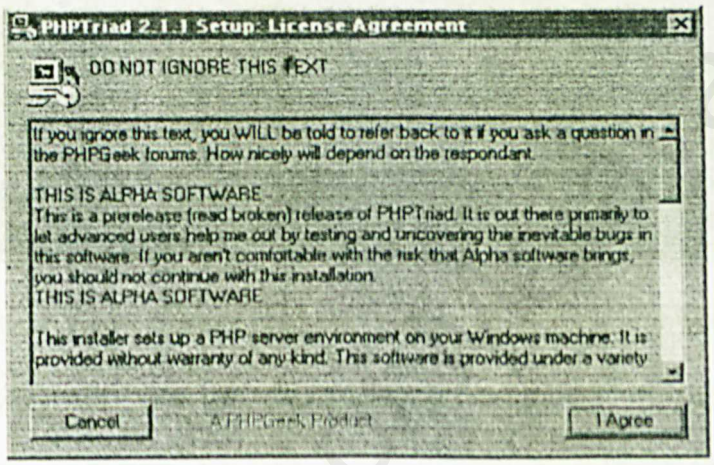
Operating System/ Web Browser	COMPATIBILITY WITH ASKAKAK.COM		
	Compatible	Not Compatible	Not Tested
Windows 95	✓		
Windows 98 SE	✓		
Windows Millennium		✓	
Windows NT4	✓		
Windows 2000	✓		
Windows XP Home Edition (32 Bit)		✓	
Windows XP Home Edition (64 Bit)		✓	
Internet Explorer Version 1-3	✓		
Internet Explorer Version 4-6			✓
Netscape 4 to Netscape 6	✓		
Opera			✓
NeoPlanet	✓		

Configuring AskAkak.Com Counselling Portal

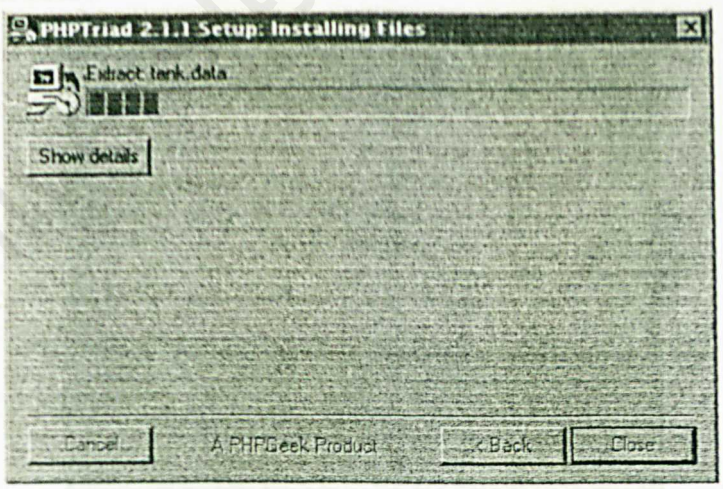
Follow the steps listed below:

Step 1: Installing PHP 4, MySQL 3.2.3 and Apache HTTP Server using PHPTriad 2-2-1

1. Insert the CD labelled AskAkak.Com Counselling Portal into your CD-ROM drive.
2. Browse through the CD.
3. Double click on the icon labelled PHPTriad 2-2-1. The window below will appear:



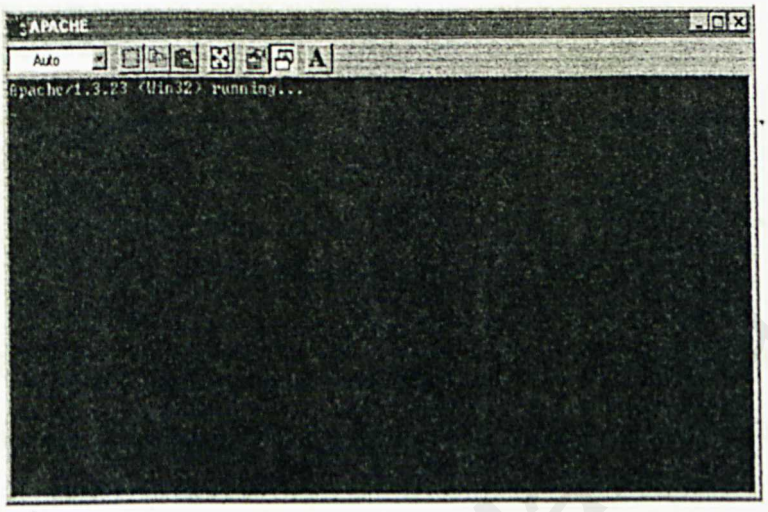
4. Click on the button 'I Agree' and wait until setup has finished installing.



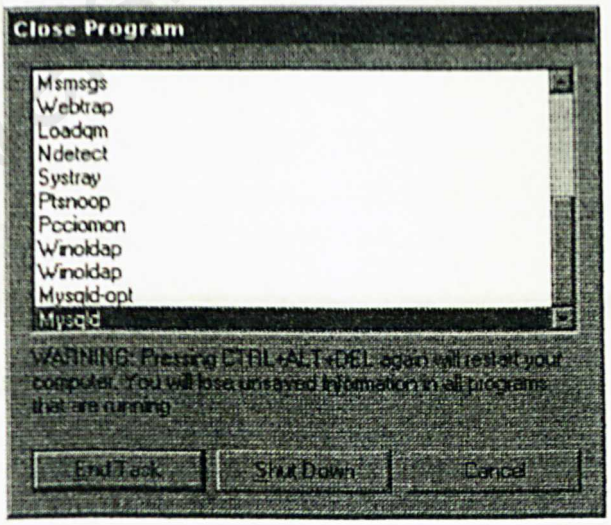
5. When setup is completed, click 'Close'.

Step 2: Setting up the database

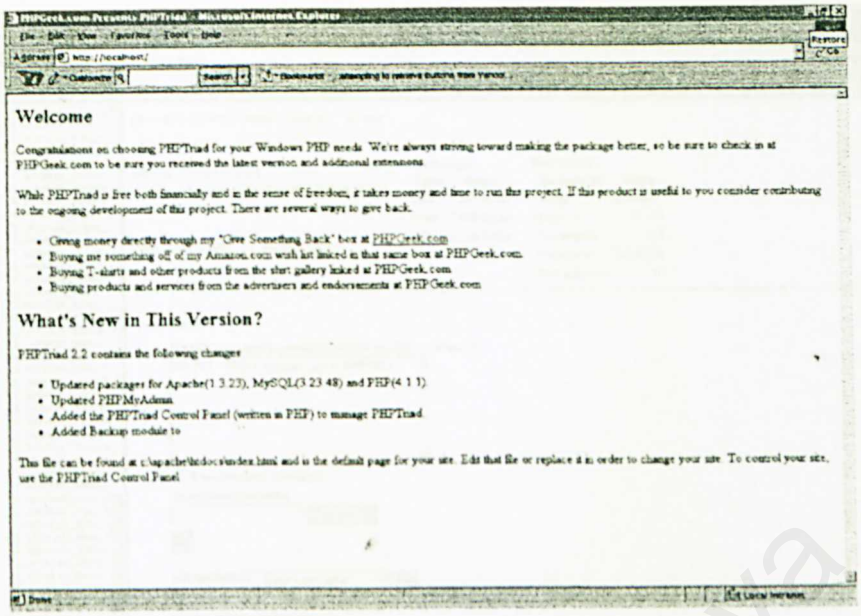
1. Go to your Start Menu. Find the Folder Apache Console in the Program PHPTriad. Click on Start Apache. The window below will appear:



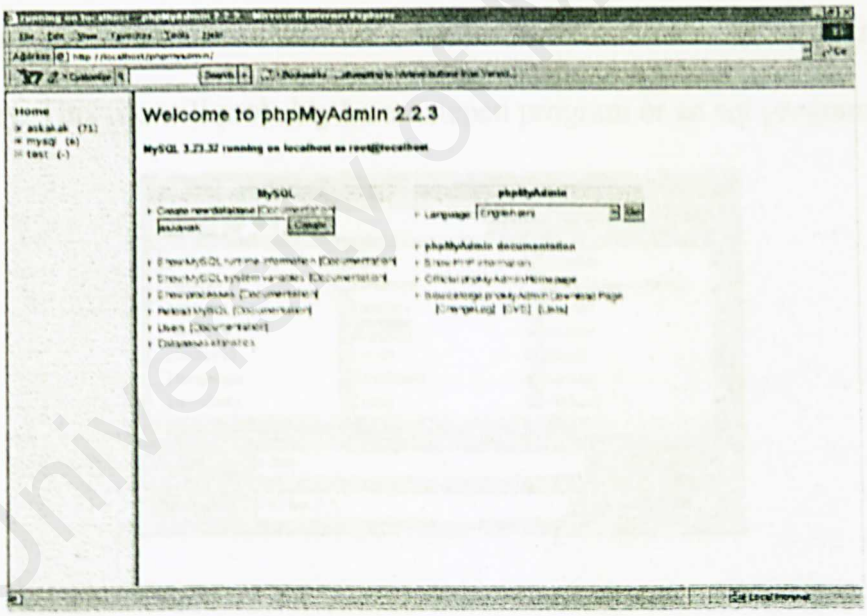
2. Go to your Start Menu Again. Find the Folder MySQL in the Program PHPTriad. Click on MySQL-D. You can check whether MySQL-D is actually running by pressing CTRL-ALT-DEL. Mysqld should be listed in the Task Manager, as shown in the diagram below:



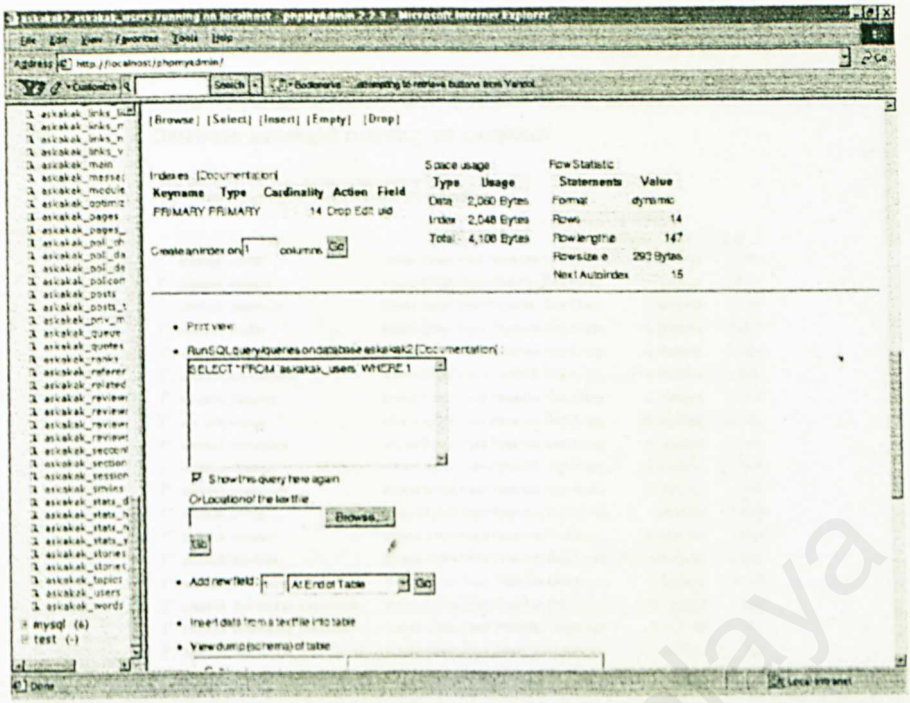
3. Open your web browser and type the following in the Address Bar: http://localhost/ . The window below will appear.



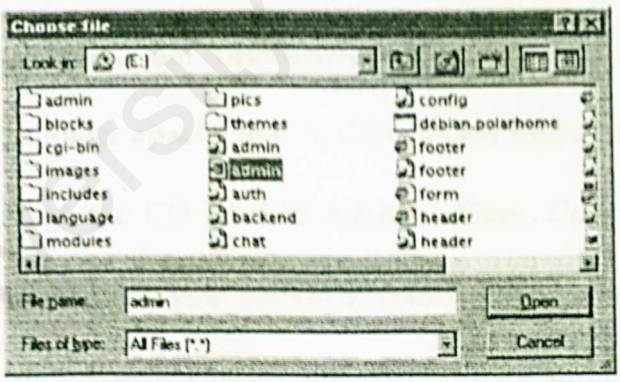
4. Next, type in http://localhost/phpmyadmin/ in the Address Bar. The screen below will appear:



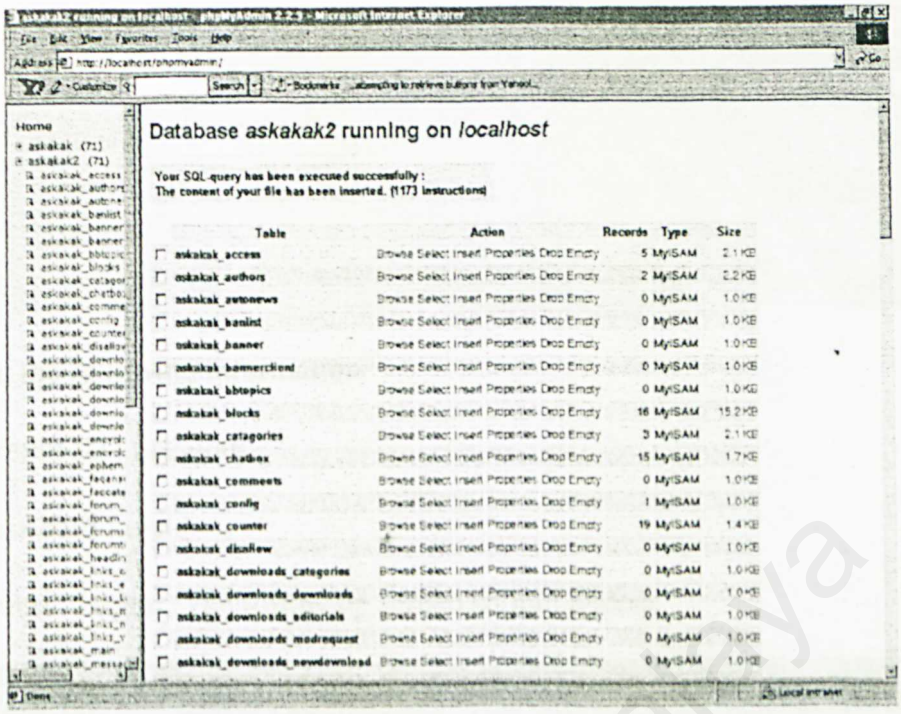
5. In the textbox Create New Database, type in **askakak**. The window below will appear.



6. Click on the button 'Browse' and click on the file admin.sql in your CD-ROM directory. This file will probably be a notepad program or an sql program.



7. Click on the button 'Go'. The database should load itself in a few minutes. After it has finished loading, the screen below will appear:



8. Congratulations! This means that your askakak database has executed successfully!

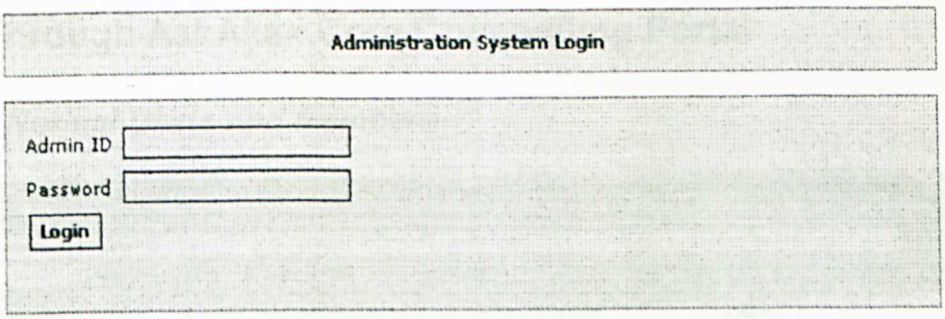
   Proceed to the next step.

Step 3: Configuring your AskAkak.Com Administration Page

1. Browse through the folder **apache** in **C:**. Click on the folder **htdocs**.
2. Browse again through the CD labelled **AskAkak.Com**. Copy the folder **askakak** from the CD and paste it into your current directory (**C:\apache\htdocs**).
3. Making sure that your Apache Console and MySQL-D is still running, launch your Internet Explorer again and type in the following in the Address Bar:

   http://localhost/askakak/admin.php

   The following screen should appear:

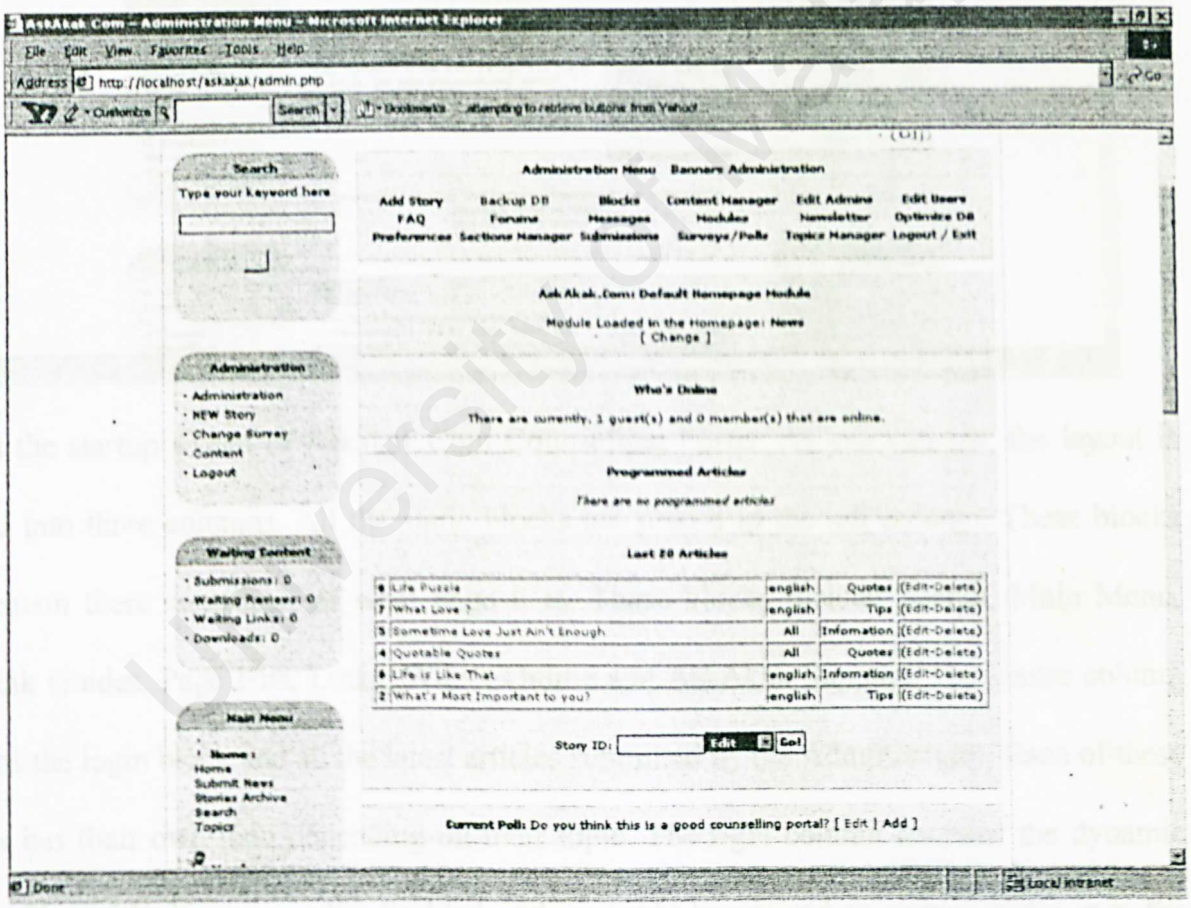


4. Key in the following information:

   Admin ID : **askakak**

   Password : **tesis**

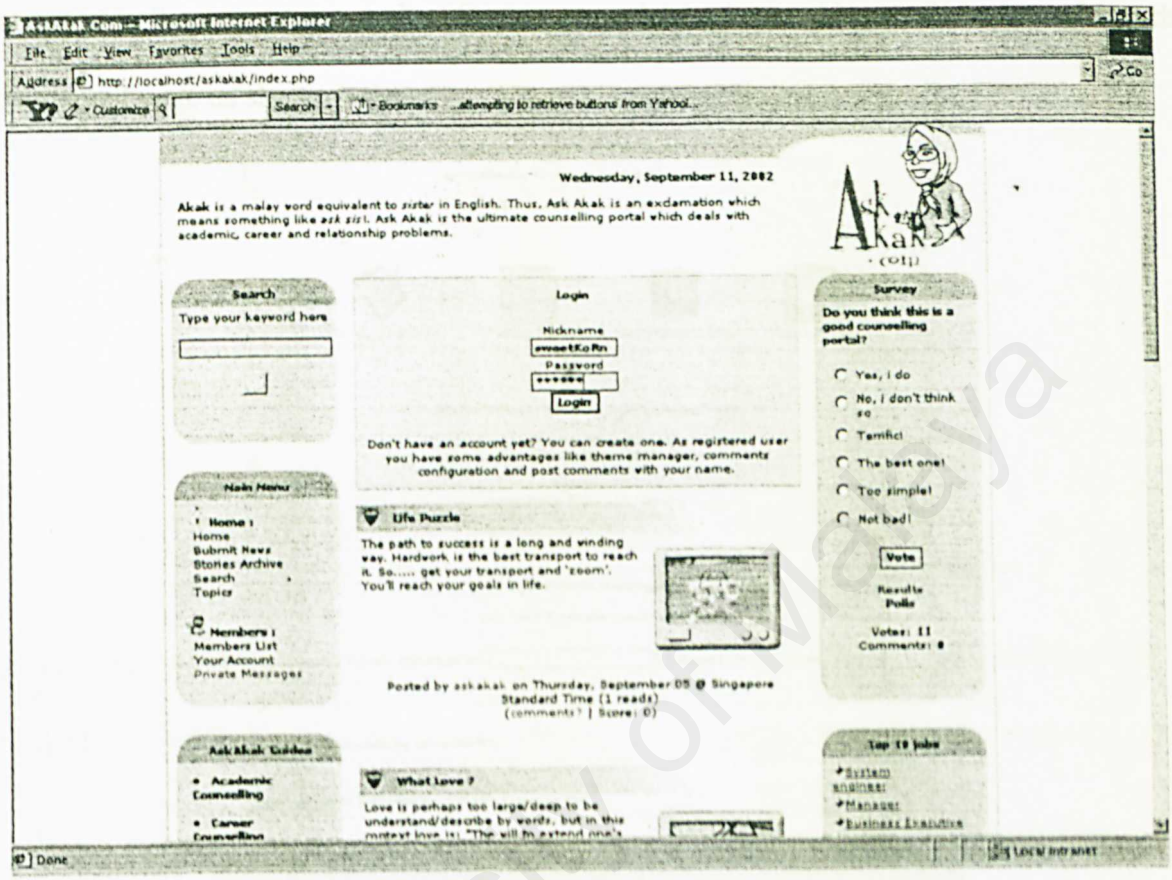
   And press 'Login'. The following screen should appear.



5. There you go! You're free to edit, delete or add anything you like. Feel free to play around and experiment with it.

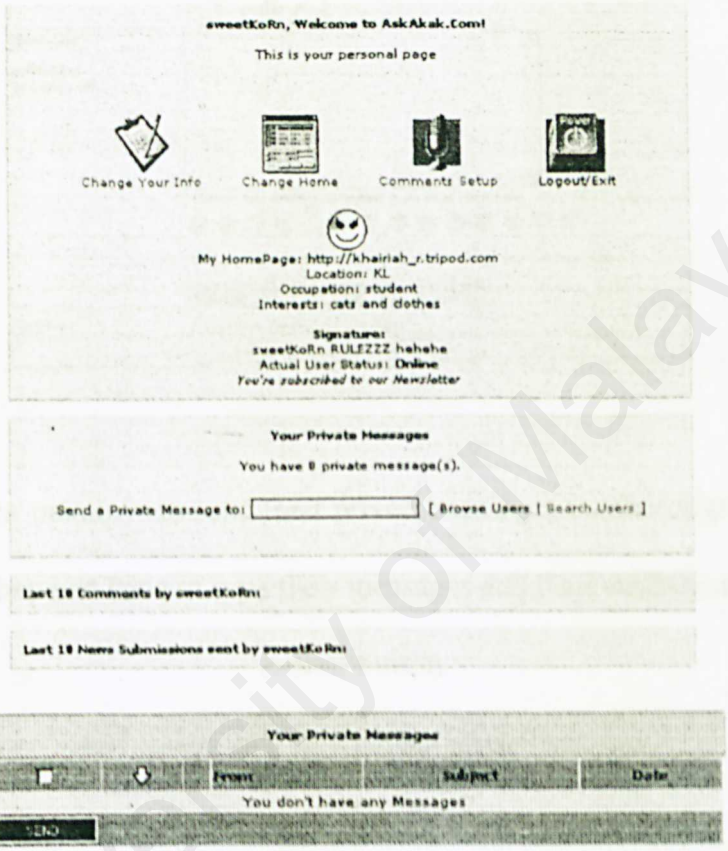
Tour Guide Through AskAkak.Com Counselling Portal

Tour Guide for Normal Users and Members

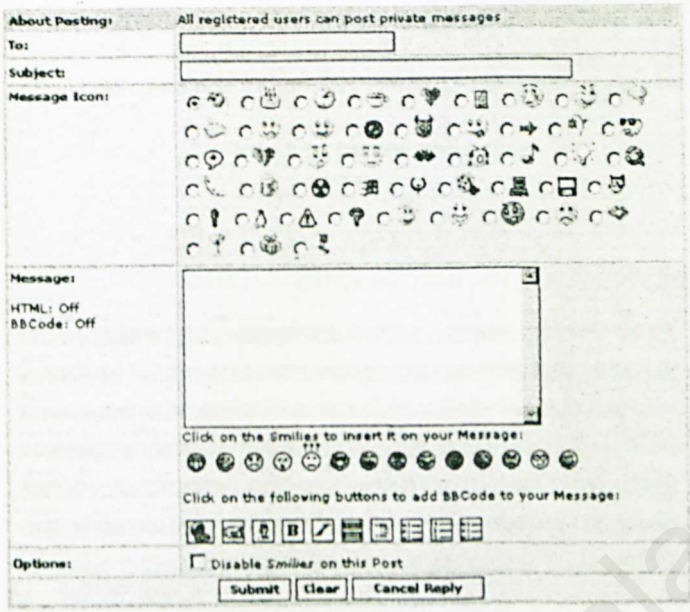


This is the startup screen of Askakak.Com Counselling Portal. As you can see, the layout is divided into three columns. All the static blocks are placed in the left column. These blocks will remain there regardless of what page it is. These blocks include Search, Main Menu, AskAkak Guides, Page Hits, Links, Who's Online and AskAkak Support. The centre column contains the login block and all the latest articles submitted by the Administrator. Each of these articles has their own icon depending on their topic. The right column contains the dynamic blocks. These blocks are usually disabled and can only be viewed in certain pages, e.g. the startup screen. These blocks are Survey, Top 10 Jobs, Perplexed!, Sources and Study Tips.

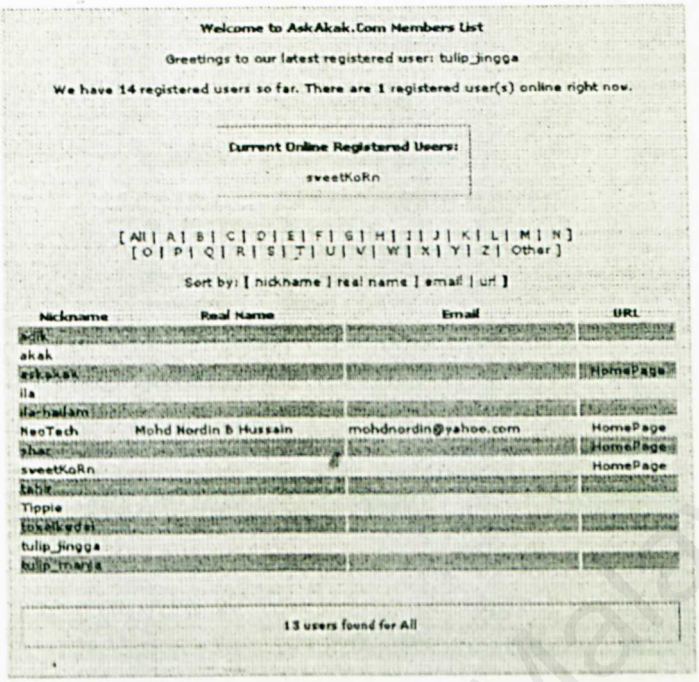
Registered users or members can login through their Nickname and Password. Normal users can also browse through the portal but their capabilities are limited, thus they are encouraged to "create an account". After a successful login, members will be brought to their own personal page as shown below:



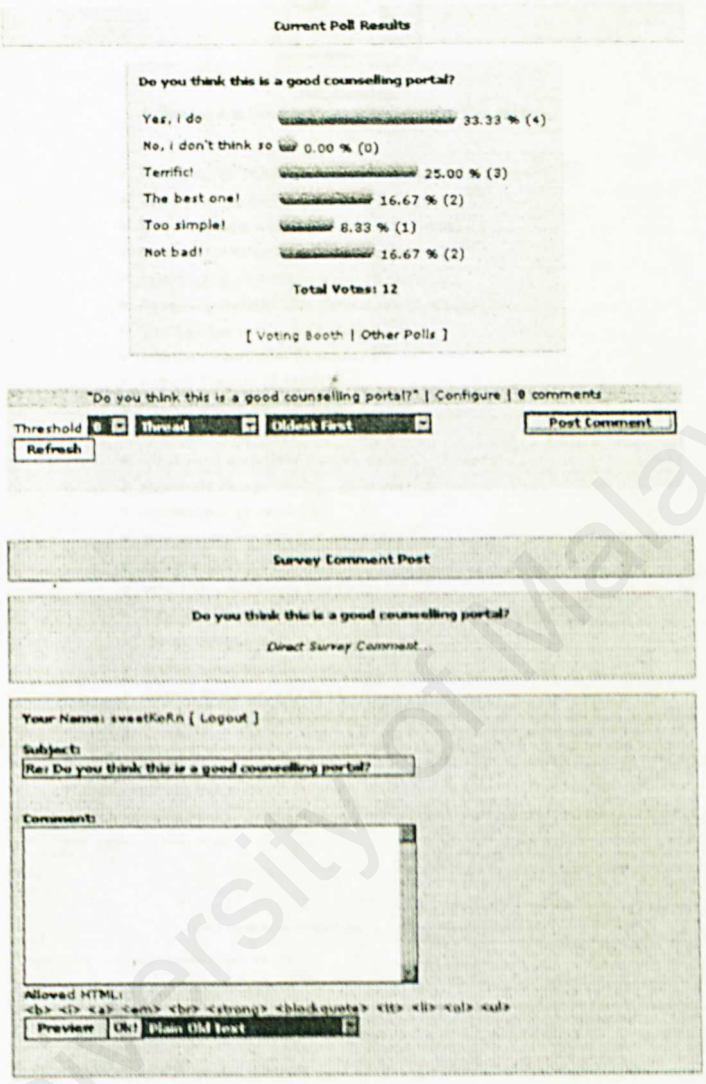
This is the *Personal Page Section* for members to change their profile.



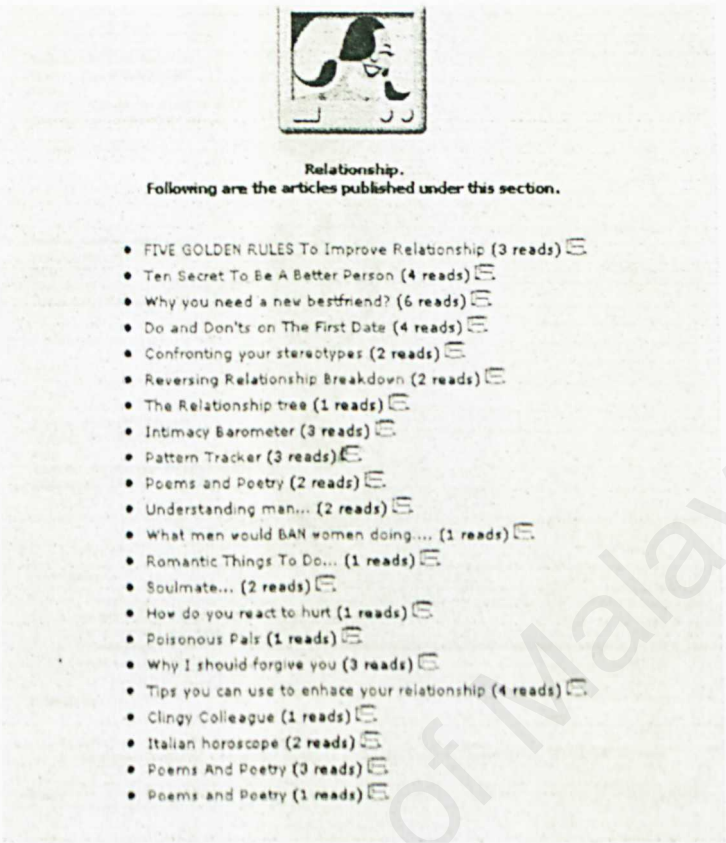
This is the section for members to send their private messages to other registered members of AskAkak.com. They just have to type their messages and then send them by clicking the *submit* button.



This is the *Member List* Section. Here, we can see all the members that have been registered under AskAkak.com. You can find their name from the list in alphabetical order.

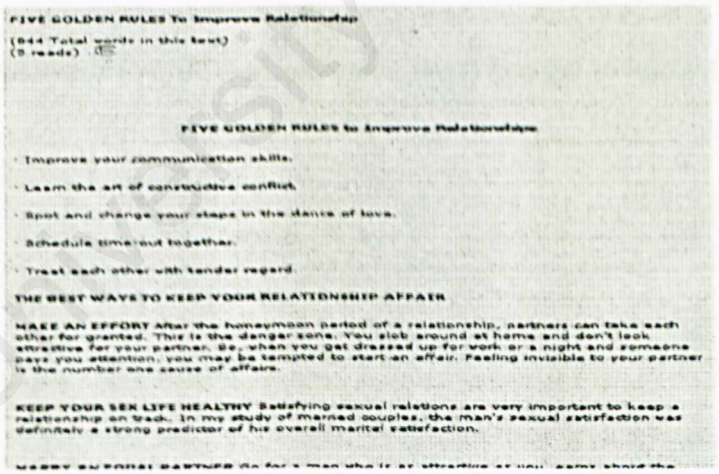


This is the section for members to give their comments about AskAkak.com. They just have to type their comments or suggestions and then send to us by clicking the *OK* button.



Relationship.
Following are the articles published under this section.

- FIVE GOLDEN RULES To Improve Relationship (3 reads)
- Ten Secret To Be A Better Person (4 reads)
- Why you need a new bestfriend? (6 reads)
- Do and Don'ts on The First Date (4 reads)
- Confronting your stereotypes (2 reads)
- Reversing Relationship Breakdown (2 reads)
- The Relationship tree (1 reads)
- Intimacy Barometer (3 reads)
- Pattern Tracker (3 reads)
- Poems and Poetry (2 reads)
- Understanding man... (2 reads)
- What men would BAN women doing.... (1 reads)
- Romantic Things To Do... (1 reads)
- Soulmate... (2 reads)
- How do you react to hurt (1 reads)
- Poisonous Pals (1 reads)
- Why I should forgive you (3 reads)
- Tips you can use to enhace your relationship (4 reads)
- Clingy Colleague (1 reads)
- Italian horoscope (2 reads)
- Poems And Poetry (3 reads)
- Poems and Poetry (1 reads)



FIVE GOLDEN RULES To Improve Relationship

(844 Total words in this text)
(3 reads)

FIVE GOLDEN RULES to Improve Relationships

- Improve your communication skills.
- Learn the art of constructive conflict.
- Spice and change your steps in the dance of love.
- Schedule time-out together.
- Treat each other with tender regard.

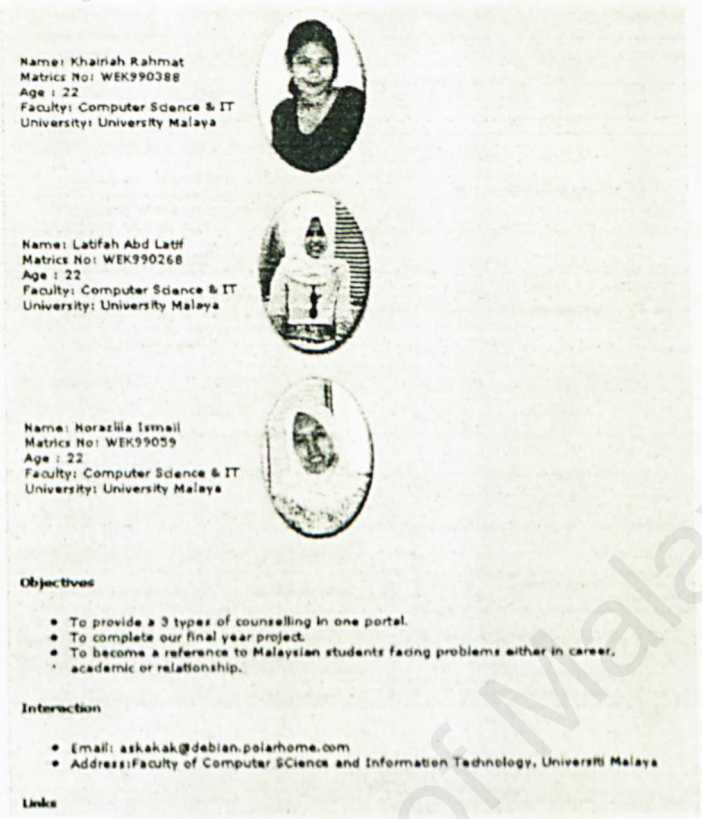
THE BEST WAYS TO KEEP YOUR RELATIONSHIP AFFAIR

MAKE AN EFFORT After the honeymoon period of a relationship, partners can take each other for granted. This is the danger zone. You slob around at home and don't look attractive for your partner. So, when you get dressed up for work or a night and someone pays you attention, you may be tempted to start an affair. Feeling invisible to your partner is the number one cause of affairs.

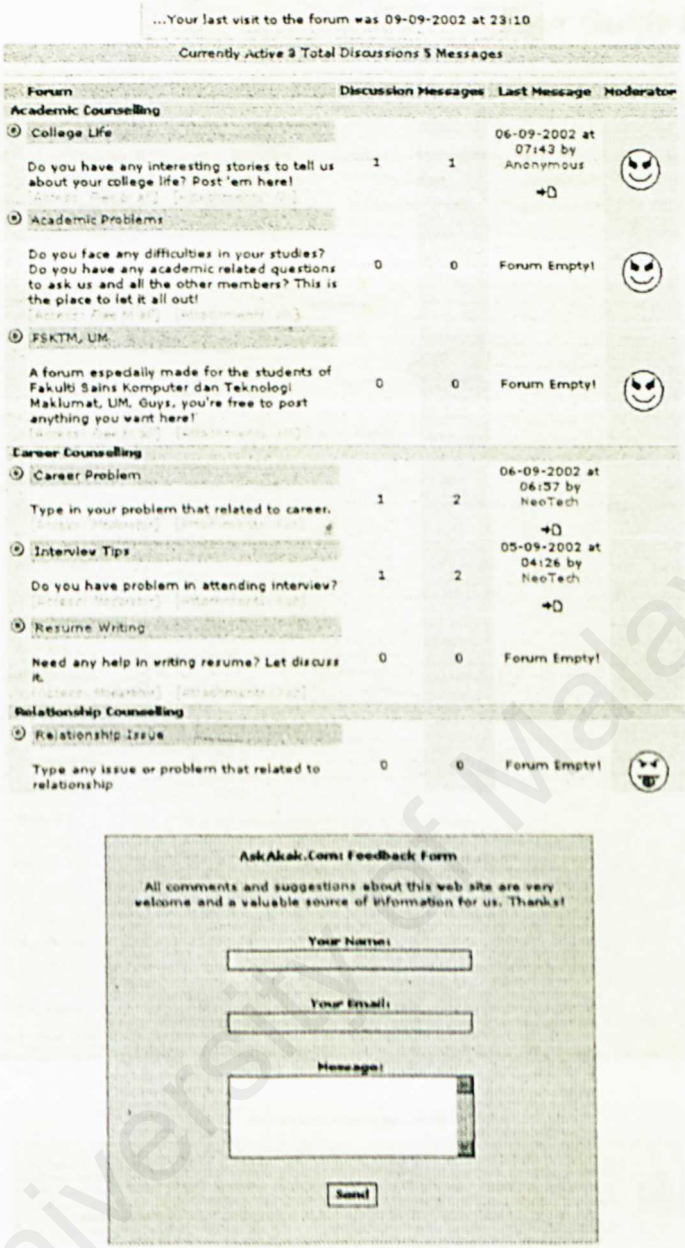
KEEP YOUR SEX LIFE HEALTHY Satisfying sexual relations are very important to keep a relationship on track. In my study of married couples, the man's sexual satisfaction was definitely a strong predictor of his overall marital satisfaction.

HAPPY EXTRAMARITAL PARTNER Go for a man who is as attractive as you. Many should the

This is the *Relationship Counselling Section.* There are 6 parts in this section such as Perplexed, Quizzes, Goodies, His/Her Say, Relationship Issue and Horoscope.

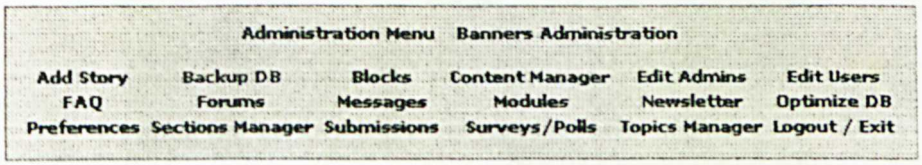


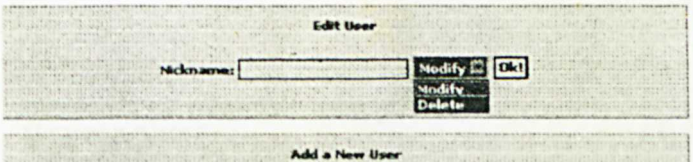
This is the *Administrator Profile Section* where users can get to know their counsellors.

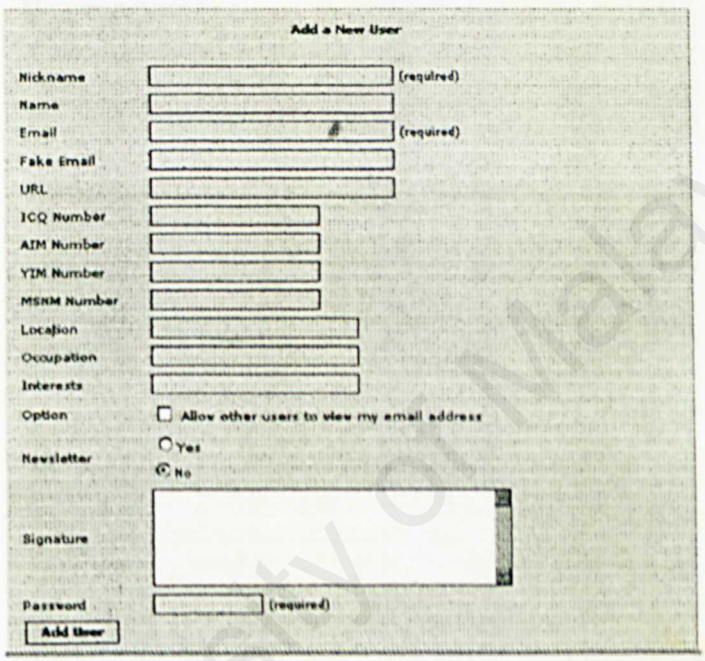


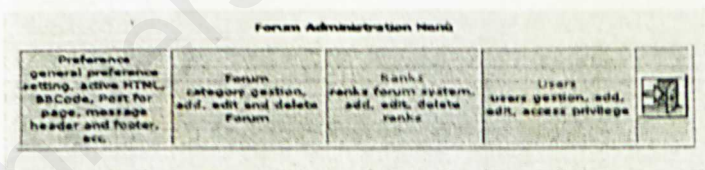
This is *the Feedback Section*. Users just have to type their message and then click the *send* button.

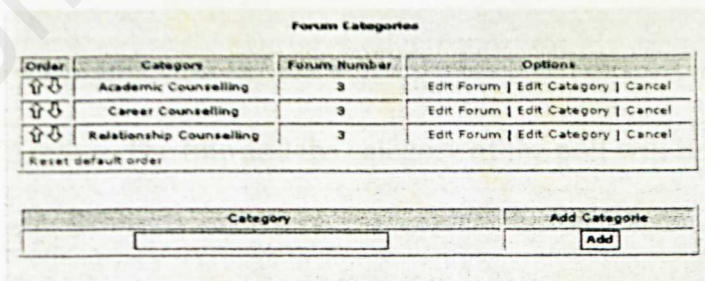
Tour Guide for Administrators



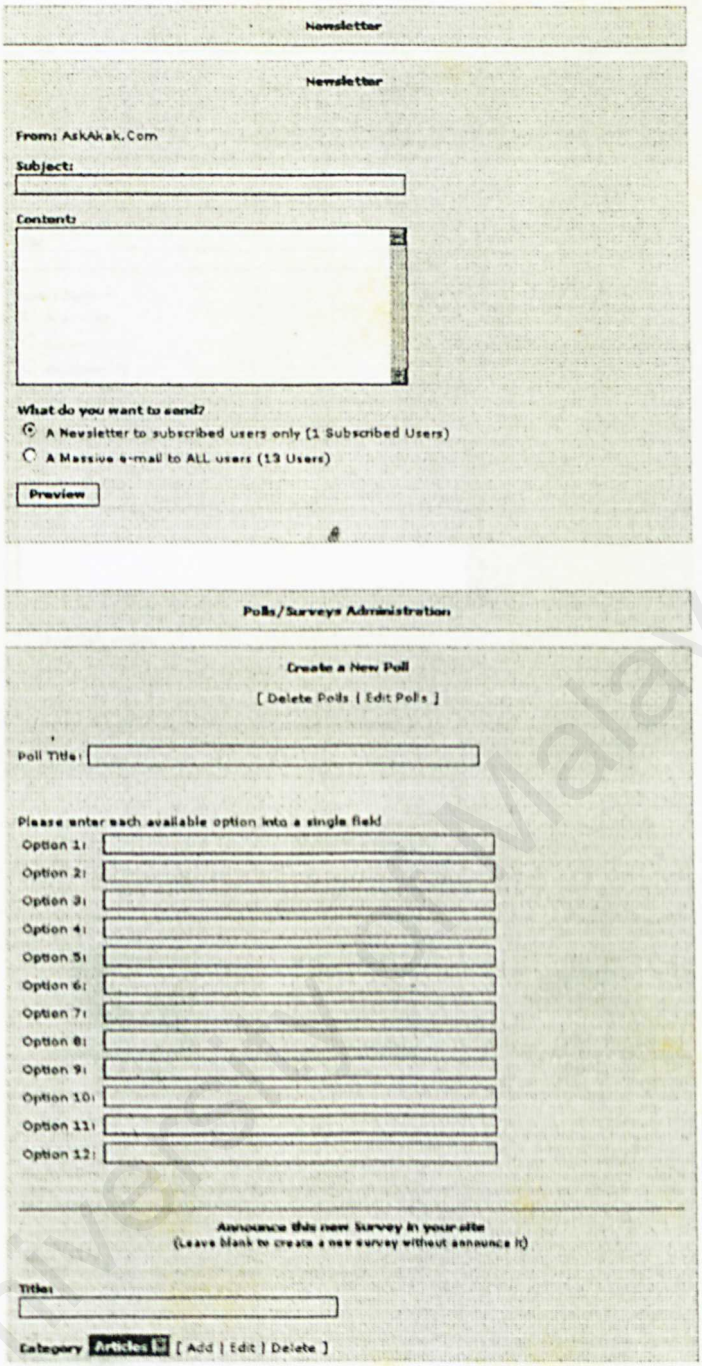




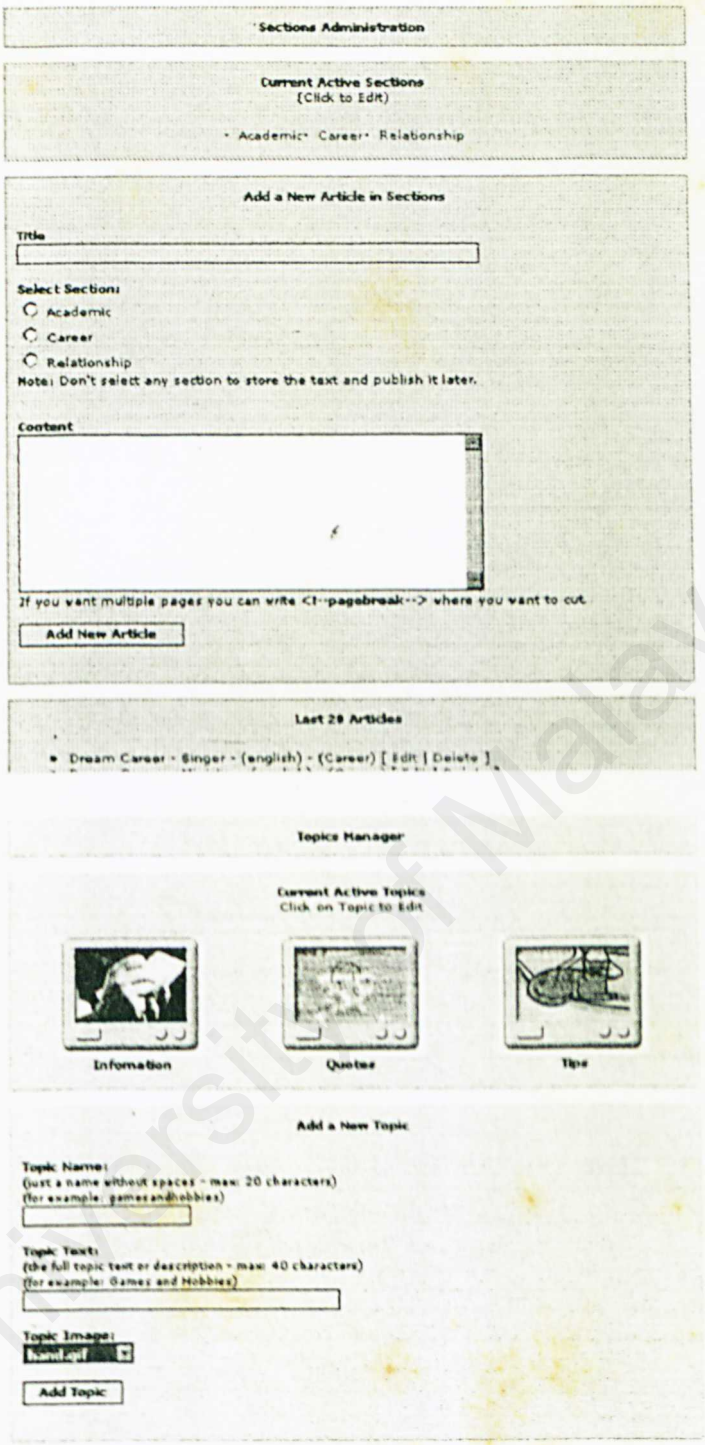




This is the *Forum Section* that especially for the administrators used. The administrators will type the relevant topic that is related to Counselling and starting the forum by clicking the *add* button.



This is the *User Poll Section.* The title and the category of the poll will be selected by the administrators.



This is the *Add New Topic section* that allows administrator to add any new topic that is related to counselling. The administrator will type the topic name, add the topic text and then click at the *add topic* button.